Robert Krimmer, Melanie Volkamer,
Véronique Cortier, David Duenas-Cid, Rajeev Goré, Manik Hapsara,
Reto Koenig, Steven Martin, Ronan McDermott, Peter Roenne, Uwe Serdült,
Tomasz Truderung (Eds.)

Third International Joint Conference on Electronic Voting

E-Vote-ID 2018

2–5 October 2018, Lochau/Bregenz, Austria

Co-organized by:

Tallinn University of Technology
Ragnar Nurkse Department of Innovation and Governance
Karlsruhe Institute of Technology
Institute of Applied Informatics and Formal Description Methods
E-Voting.CC GmbH
Competence Center for Electronic Voting and Participation
Gesellschaft für Informatik
German Informatics Society, SIG SEC/ECOM
Kastel
Competence Center for Applied Security Technology

PROCEEDINGS

Robert Krimmer, Melanie Volkamer,
Véronique Cortier, David Duenas-Cid, Rajeev Goré, Manik Hapsara, Reto Koenig, Steven Martin, Ronan McDermott, Peter Roenne, Uwe Serdült, Tomasz Truderung (Eds.)

Third Joint International Conference on Electronic Voting

E-Vote-ID 2018

2–5 October 2018, Lochau/Bregenz, Austria

Co-organized by the Tallinn University of Technology, Karlsruhe Institute of Technology, E-Voting.CC, Gesellschaft für Informatik and Kastel

TTÜ PRESS

Proceedings E-Vote-ID 2018
TalTech Press

ISBN 978-9949-83-335-1

Volume Editors

Prof. Dr. Dr. Robert Krimmer
Tallinn University of Technology
Ragnar Nurkse Department of Innovation and Governance
Akadeemia tee 3
12618 Tallinn
Estonia
robert.krimmer@taltech.ee

Prof. Dr. Melanie Volkamer
Karlsruhe Institute of Technology
Institute of Applied Informatics and Formal Description Methods
Kaiserstr. 89
76131 Karlsruhe,
Germany
melanie.volkamer@secuso.org

Dr. Véronique Cortier
Laboratoire Lorrain de Recherche en
Informatique et ses Applications, France
E-mail**:** veronique.cortier@loria.fr

Dr. David Duenas-Cid
Tallinn University of Technology, Estonia
E-mail: david.duenas@ttu.ee

Prof. Dr. Rajeev Goré
Australian National University, Australia
E-mail: rajeev.gore@anu.edu.au

Dr. Manik Hapsara
University of New South Wales, Australia
E-mail: hapsara.manik@gmail.com

Reto Koenig
Bern University of Applied Sciences
E-mail: reto.koenig@bfh.ch

Steven Martin
OSCE - Office for Democratic
Institutions and Human Rights, Poland
E-mail: steven.martin@odihr.pl

Ronan McDermott
Independent Elections and Technology
Consultant, Switzerland
E-mail: ronan@mcdis.com

Dr. Peter Roenne
University of Luxembourg, Luxembourg
E-mail: peter.roenne@gmail.com

Prof. Dr. Uwe Serdült
Ritsumeikan University, Japan / Centre
for Democracy Studies, Switzerland
E-mail: uwe.serdult@zda.uzh.ch

Dr. Tomasz Truderung
Polyas, Germany
E-mail: t.truderung@polyas.de

printed by TalTech Press, Tallinn (Estonia)

This conference is co-organized by:

Tallinn University of Technology - Ragnar Nurkse Department of Innovation and Governance

Karlsruhe Institute of Technology - Institute of Applied Informatics and Formal Description Methods

E-Voting.CC GmbH - Competence Center for Electronic Voting and Participation

Gesellschaft für Informatik, German Informatics Society, SIG SEC/ECOM

Kastel, Competence Center for Applied Security Technology

Supported by:

Regional Government of Vorarlberg

Swiss Federal Chancellery

General Chairs

Krimmer, Robert (Tallinn University of Technology - Ragnar Nurkse Department of Innovation and Governance, Estonia)
Volkamer, Melanie (Karlsruhe Institute of Technology - Institute of Applied Informatics and Formal Description Methods)

Track on Security, Usability and Technical Issues

Cortier, Véronique
Laboratoire Lorrain de Recherche en Informatique et ses Applications, France
Goré, Rajeev
Australian National University, Australia

PhD Colloquium

Koenig, Reto
Bern University of Applied Sciences, Switzerland
Tomasz Truderung
Polyas, Germany

Track on Administrative, Legal, Political and Social Issues

Hapsara, Manik
University of New South Wales, Australia
Serdült, Uwe
Ritsumeikan University, Japan / Centre for Democracy Studies, Switzerland

Local Organization Committee

Duenas-Cid, David
Tallinn University of Technology, Estonia
Krivonosova, Iuliia
Tallinn University of Technology, Estonia
Traxler, Gisela
E-Voting.CC, Austria
(Main Contact)

Track on Election and Practical Experiences

Martin, Steven
OSCE - Office for Democratic Institutions and Human Rights, Poland
McDermott, Ronan
Independent Elections and Technology Consultant, Switzerland

Outreach Chairs

Duenas-Cid, David
Tallinn University of Technology, Estonia
Rønne, Peter
University of Luxembourg, Luxembourg

Preface

This volume contains papers presented at E-Vote-ID 2018. The Third International Joint Conference on Electronic Voting, held during October 2–5, 2018, in Bregenz, Austria. It resulted from the merging of EVOTE and Vote-ID.

More than 800 experts from over 35 countries have attended the conference series over the last 14 years. This shows that the conference continues to be one of the major events in the field of electronic voting, providing ample room for interdisciplinary and open discussion of all issues relating to electronic voting.

Also, this year, the conference consisted of:
– Security, Usability and Technical Issues Track
– Administrative, Legal, Political and Social Issues Track
– Election and Practical Experiences Track
– PhD Colloquium on the day before the conference
This year's edition, E-VOTE-ID 2018, received 45 submissions, being, each of them, reviewed by 3 to 4 program committee members, using a double blind-review process. As a result, 24 papers were accepted for the conference, plus 7 papers for the PhD Colloquium. The selected papers cover a wide range of topics connected with electronic voting, including experiences and revisions of the real uses of E-voting systems and corresponding processes in elections.

Special thanks go to the members of the international program committee for their hard work in reviewing, discussing, and shepherding papers. They ensured the high quality of these proceedings with their knowledge and experience. We would also like to thank the German Informatics Society (Gesellschaft für Informatik) with its ECOM working group for their partnership over many years.

October 2018
Bregenz, Austria

Robert Krimmer
Melanie Volkamer
Véronique Cortier
David Duenas-Cid
Rajeev Goré
Manik Hapsara
Reto Koenig
Steven Martin
Ronan McDermott
Peter Roenne
Uwe Serdült
Tomasz Truderung

Table of Contents

Managerial and Legal Issues

Improving Models

Protocols

Lessons Learned

Evaluating Practical Experiences

Counting functions — Blockchain

Risk-Limiting Audits

Attacks

PhD Colloqium

Demo Session

Program Committee

Myrto Arapinis	The University of Edinburgh
Roberto Araujo	Universidade Federal do Pará (UFPA)
Jordi Barrat i Esteve	eVoting Legal Lab
Josh Benaloh	Microsoft
David Bismark	Votato
Nadja Braun Binder	University of Zurich
Christian Bull	The Norwegian Ministry of Local Government and Regional Development
Susanne Caarls	Election Consultant
Gianpiero Catozzi	UNDP
Véronique Cortier	CNRS, Loria
Stephanie Delaune	IRISA
Ardita Driza Maurer	self-employed; Zentrum für Demokratie Aarau/Zurich University
David Duenas-Cid	Tallinn University of Technology
Aleksander Essex	University of Western Ontario
Joshua Franklin	NIST
David Galindo	University of Birmingham
Micha Germann	Katholieke Universiteit Leuven
J Paul Gibson	Mines Telecom
Kristian Gjøsteen	Norwegian University of Science and Technology
Nicole Goodman	University of Toronto
Rajeev Goré	Australian National University
Ruediger Grimm	University of Koblenz
Rolf Haenni	Bern University of Applied Sciences
Thomas Haines	Queensland University of Technology
Thad Hall	MPR
Manik Hapsara	The University of New South Wales
Toby James	University of East Anglia
Tarmo Kalvet	Ragnar Nurkse Department of Innovation and Gover-nance, Tallinn University of Technology
Norbert Kersting	University of Münster
Aggelos Kiayias	National and Kapodistrian U. Athens
Shin Kim	Hallym.University
Reto Koenig	University of Applied Sciences Berne
Robert Krimmer	Tallinn University of Technology, Ragnar Nurkse School of Innovation and Governance
Ralf Kuesters	University of Stuttgart
Oksana Kulyk	TU Darmstadt
Dan Malinovich	UNDP
Steven Martin	OSCE/ODIHR
Anu Masso	ETH Zurich
Ronan McDermott	Independent Election Expert

Juan Manuel Mecinas	Centro de Investigación y Docencia Económica
Hannu Nurmi	University of Turku
Jon Pammett	Carleton University
Olivier Pereira	Universite catholique de Louvain
Goran Petrov	Independent Election Expert
Marco Prandini	UNIVERSITA DI BOLOGNA
Josep Mª Reniu	University of Barcelona
Peter Roenne	SnT, University of Luxembourg
Mark Ryan	University of Birmingham
P. Y. A. Ryan	University of Luxembourg
Steve Schneider	University of Surrey
Berry Schoenmakers	Eindhoven University of Technology
Carsten Schuermann	IT University of Copenhagen
Uwe Serdült	University of Zurich
Vanessa Teague	Melbourne School of Engineering
Tomasz Truderung	Polyas
Priit Vinkel	State Electoral Office of Estonia
Melanie Volkamer	Karlsruhe Institute of Technology
Kåre Vollan	Quality AS
Bogdan Warinschi	University of Bristol
Roland Wen	The University of New South Wales
Gregor Wenda	BMI
Peter Wolf	International IDEA
Michael Yard	IFES
Filip Zagorski	Wroclaw University of Technology

Managerial and Legal Issues

Winning the Election, but Losing the Litigation: A Prognosis of Nigerian Judicial Attitudes toward Evidence Produced from 'E-Accreditation Machines'

Felix Oludare, Omosele✉ [0000-0002-2752-3798]

Leuphana Universitaet, Lüneburg, Germany
felix.omosele@gmail.com

Abstract.

There is already a developed body of literature on electronic accreditation in the context of elections, but discourse on the evidence-related consequences of this e- accreditation is sparse.

In other words, scholarship's searchlight on the admissibility and weight of "computer evidence from e-accreditation machines (CEEM)" in electoral litigation needs to be brighter than ever. This legal inquiry is important as elections are sometimes won or lost in courts and on the basis of such electronically-obtained electoral data. This gap will be filled by the present paper.

Using Nigeria as a case study, this paper undertook a doctrinal analysis of its appellate courts' opinions on "electronic accreditation". The analysis shows that the admissibility of CEEM often turns on the age-long hearsay rule; however, considering the unique nature of CEEM, this paper argues for a revised attitude towards the hearsay rule.

In ascribing weight to the admitted CEEM, this paper proposes a standard that is based on the Relative Plausibility Theory; this standard will ensure that parties' evidence is fairly evaluated and that winners are not made into losers in the courts.

Keywords: e-registration, e-accreditation, evidential weight, computer evidence, hearsay, Nigeria.

1 Introduction

One of the consequences of globalization is that international norms and standards are now being prescribed for sectors that have historically had a nationalistic outlook. This holds true for the electoral systems of most sovereign nations, particularly those from developing democracies in sub-Saharan Africa. Thus, the concept of electoral integrity is quickly influencing the electoral processes of these democracies.

The importance of electoral integrity has received some attention in the literature. According to Norris [1], properly conducted elections help to elect public office holders and confer legitimacy upon elected authorities. Furthermore, while there are divergent theories of democracy, there appears to be a consensus that flawed elections are injurious to the sustainment of democratic values [1]. The need for improved election administration, preservation of the integrity of elections, and the enfranchisement of the visually impaired necessitated the development and usage of electronic voting machines [2].

In recent times, developing democracies- like those of Nigeria[1]- have started utilizing some basic forms of voting technology[2] (i.e., e-identification) and this development has given rise to several statutory, evidential, and technical considerations. Therefore, the present use of these voting technologies, including the foreseeable deployment of EVM, and the impacts of evidence derived from these technologies in electoral litigations justify a critical analysis of some fundamental legal challenges.

Furthermore, though the literature (see e.g. [3, 4, 5]) has highlighted the constant conflict between e-voting codes and higher-order statutes[3], the discourse has, however, not adequately captured the hearsay and other evidential implications of these voting technologies and their outputs in electoral litigation. This paper will fill this gap by drawing on lessons from the few decided Nigerian judicial precedents that border on the application of e-accreditation rules in the electoral context.

Perhaps the most recurrent issue with the use of "Computer Evidence from Electronic-accreditation Machines" (CEEM)[4] in electoral litigation in Nigeria is the hearsay[5] challenge. This paper will review the statutory and judicial attitudes toward this

[1] Nigeria's choice is strategic. Nigeria represents a group of countries with manual voting systems augmented by the most basic form of voting technology. Therefore, an incisive analysis of the evidential challenges of "evidence from e-accreditation machines" in these countries will provide their judicial systems with a good head-start for when and if they transit to full-blown electronic voting.

[2] At present, Nigeria utilizes voting technology for the purposes of voters' registration and accreditation. These technologies are in the form of a "direct data capture (DDC)" device for voters' registration and an "electronic card reader device" for the purpose of voters' accreditation.

[3] For example, conflict between e-accreditation rules on the one hand and constitutionally guaranteed rights or rights enshrined in the Electoral Act on the other hand.

[4] This species of evidence will hereinafter be simply referred to as "CEEM" in this paper.

[5] The "hearsay rule" is a long-standing rule in the law of evidence. Subject to established exceptions, it states that only the maker of a statement is permitted to tender such statement

challenge for e-accreditation. It will therefrom argue that a relaxed attitude toward hearsay with respect to CEEM is commendable and that judges should be more concerned with ascertaining the weight ascribable to CEEM.

Building up from there, the paper will discuss how courts can ascribe weight to the admitted "hearsay" CEEM. This discussion will be normative and based on the underlying principles of the Relative Plausibility Theory (RPT) [6]. The paper will contend that this normative standard, if followed, will ensure that the evidential goals of factual accuracy and fair allocation of risk of errors in trials are achieved (see the arguments of Pardo on this in Section 2.1).

This paper will proceed as follows. Section 2 will provide an overview of the theoretical and conceptual framework that will be adopted in the paper. A doctrinal analysis[6] of judicial decisions will be undertaken in section 3 to highlight the potential consequence of e-accreditation rules on extant electoral statutory provisions. In section 4, the hearsay challenge to the admissibility of CEEM will be discussed and a normative framework for evaluating the weight of CEEM will be proffered. The concluding section will proffer some recommendations.

2 Theoretical and Conceptual Framework

2.1 Theoretical Framework

Pardo [7] argued that a successful evidentiary theory must accord with two basic epistemological foundations: factual accuracy and allocation of the risk of factual errors. "Factual accuracy" means that a theory must be able to account for a sufficient level of accuracy in its outcomes, whereas the "allocation of risk of errors" relates to an equal treatment of the parties when called upon to persuade the court as to their claims [7]. In other words, a sound theory of evidence must be founded on the necessary foundation of being accurate and fair. One such theory is the RPT.

The Relative Plausibility Theory[7] (in the form developed by Professor Ronald J. Allen) explains that evidence is the result of the interaction of the intelligence and knowledge of the fact finder coupled with the sum of the observations captured during a trial [6]. As such, this paper will be guided by the RPT because it satisfies both of the necessary conditions: it strives for accuracy and allocates the risk of factual errors.

The theory is made up of two sub-theories, namely the structural theory of juridical proof and the theory of juridical evidence. We shall now proceed to examine each of the sub-theories.

in proof of the truth thereof. The general rule is stated in section 38 of the Nigerian Evidence Act.

[6] The Doctrinal Method was adopted because it allows this researcher to interpretatively analyze judicial reasoning in electoral litigations.

[7] Hereinafter referred to as the "RPT."

2.1.1 Structural Theory of Juridical Proof

This sub-theory deals with the formal structure of the proof process itself. It relates to what is to be proved (e.g., the elements of a civil cause) and the requisite standard of proof (e.g., the preponderance of evidence in a civil cause) [6].

Therefore, for the structural theory, "what is to be proven" is a story or set of stories that must be "told" as being more plausible than its competitors. Here, the task of the fact finder (a jury or a trial judge) is to determine the relative plausibility of the parties' stories and then allocate whatever ambiguities exist equally across the parties.

2.1.2 Theory of Juridical Evidence

This sub-theory of the RPT recognizes that there are three broad types of evidence: oral evidence, physical evidence, and miscellaneous trial observations. It defines evidence as not being a set of things but the process by which fact-finders come to conclusions about the past [6].

However, as a departure from the conventional theory of evidence, this sub-theory ascribes much value to the fact-finder's experiences at the moment of the decision, experiences which should affect the materials (oral and physical evidence), and other observations generated at trial [6]. In essence, the theory predicts that judicial decision makers' reasoning should be explanation-based and that rules should only be used to justify the outcome reached.

2.2 Conceptual Framework

A conceptualization of the key terms in the research question now follows.

2.2.1 Computer Evidence

In this paper, the term "evidence"- except where otherwise stated- is used in the loose sense, i.e., the term corresponds more to "data" or "records" that are capable of being legally admitted in proof of some facts. Therefore, "computer evidence" refers to "electronically stored information" (ESI) that is capable of being offered in court as proof of some facts- as opposed to an "electronically generated record" that is solely the creation of a computer and is therefore not subject to the hearsay rule [8].

In its broader sense, however, "evidence," in line with the RPT, will be seen as a process by which fact-finders come to conclusions about the past [6]. It will thus include not only physical evidence (e.g., reports of electronically-stored accredited voters), but also oral evidence and miscellaneous observations generated at trial. This conceptualization will guide this paper's central argument that triers of facts should play greater roles in interpreting hearsay and evaluating the weight ascribable thereto.

2.2.2 Electronic Accreditation Machines and E-accreditation

The conceptualization of electronic accreditation machines (EAM) in this research is limited to machines that are used for electronic voter registration prior to Election Day and voters' accreditation on Election Day. Therefore, e-accreditation is the use of electronic means to register and accredit voters but not using those means for actual voting purposes.

Thus, while the "Council of Europe Legal Standards for e-voting", defines e-voting as "the use of electronic means to cast and/or count the vote" [9], this paper's conceptualization embraces only "e-accreditation". In this regard, the paper's findings and recommendations are confined to e-accreditation and EAM.

However, e-accreditation, as conceptualized here, will exclude the use of the electronic machines in "uncontrolled environments" (e.g., voters' registration and accreditation over the internet) and will be limited to their usage in "controlled environments" (e.g., approved polling stations). See generally the discussion on the definition of e-voting in [3].

2.2.3 "Computer Evidence from E-accreditation Machines" (CEEM).

CEEM, as conceptualized in this research, relates to computer evidence from EAM.

3 Judicial Opinions from Nigeria

In Nigeria, the decision of the Supreme Court in the case of *Nyesom & Others v. Peterside & Others*[10] brought to the fore the issue of how e-accreditation and its legal regime sometimes result in an "overreaching" of other higher-order statutes, like the Electoral Act.

To this end, this section will analyze how Nigerian appellate courts have interpreted these statutes and have successfully "isolated" the overreaching effect of any applicable e-accreditation rules.

3.1 Voting Technology, Electoral Rules, and Frictions with Higher-order Statutes in Nigeria

Prior to the analysis of these judicial decisions, it will be worthwhile to briefly discuss the relevant Nigerian standards and rules governing electoral e-registration, e-accreditation, and their admissibility in election petitions.

3.1.1 E-registration and Accreditation

Despite the fact that "electoral legislation" falls under the "concurrent list" in the second schedule of the Nigerian Constitution [11], the Constitutional structure, nonetheless, gives the National Assembly a pre-eminent position in electoral legislation.

Therefore, though the States' Legislative Houses can "… [make] laws with respect to election to a local government council…" Part II, paragraph 11 of the second schedule still subject their powers to that of the National Assembly. Therefore, the National Assembly can "make laws for the Federation with respect to the registration of voters and the procedure regulating elections *(even)* to a local government council..."[11]

Thus, the Electoral Act, a federal legislation, has been, unsurprisingly, at the center of most notable election petition holdings in Nigeria. With the establishment of the

Independent National Electoral Commissions (INEC) by section 153(1)f of the Constitution [11] and the delegation of subsidiary rule-making to it in the Electoral Act [12], the coast was clear for the INEC to issue the much litigated ‘2015 Election Guidelines’.

While the e-registration of voters prior to Election Day- is clearly stipulated by INEC on its website [13], paragraphs 8-15 of the Electoral Guidelines provided the standards to be observed by electoral officers for the e-accreditation of voters on Election Day [14].

3.1.2 Higher-Order Statutory Provisions Regulating Admissibility of Electronic Evidence and E-accreditation

Sections 84(1) and (2) of the Evidence Act of 2011 [15] innovatively provide for the recognition of electronic evidence (and by implication, CEEM) and lay down conditions for the admissibility of computer-generated documentary statements. These conditions, if met, will provide proof of the chain of custody and reliability of the CEEM prior to being tendered in evidence. The section provides thus:

> (1) In any proceeding a statement contained in a document produced by a computer shall be admissible as evidence of any fact stated in it of which direct oral evidence would be admissible. *if it is shown that the conditions in subsection (2) of this section are satisfied in relation to the statement and computer in question.*
> (2) The conditions referred to in subsection (l) of this section are-
> (a) that the document containing the statement was produced by the computer during a period over *which the computer was used regularly to store or process information* for the purposes of any activities regularly carried on over that period…;
> …
> (c) that throughout the material part of that period *the computer was operating properly* or, if not, that in any respect in which it was not operating properly or was out of operation during that part of that period was not such as to affect the production of the document or the accuracy of its contents; …

If applied to this research focus, the “document” referred to in section 84(1) will be the “Card Readers’ Voters Accreditation Report,” whereas the “computer” will be the “card reader” device itself.

Therefore, to authenticate any form of electronic evidence under the Act, the state of the electronic device at the time of storing/producing the data is as important as the eventual evidence (data) produced. Thus, to prove that the conditions stated in section 84(2) have been complied with, the proponent of the electronic evidence is required by section 84(4)b to produce a certificate signed by a person occupying “…a responsible position in relation to the operation of the relevant device…”[15].

3.1.3 Conformity of E-accreditation Rules to Extant Constitutional and Statutory Provisions

Section 52(2) of the Electoral Act provides that electronic voting is prohibited for the time being in Nigeria [12] but voter registration and accreditation via electronic devices is permitted by paragraph 8b of the 2015 Electoral Guidelines [14].

However, prior to the 2015 general elections, while the principal legislation- the Electoral Act- stipulated that voters' accreditation on voting day was to be determined using a manual voters' register, the Electoral guidelines provided for the determination of the same using an electronic card reader device. Unsurprisingly, the situation amounted to a "waiting time bomb" for electoral management and was only resolved by the intervention of the judiciary in a host of cases.

For example, in the Supreme Court case of *Nyesom & Others v. Peterside & Others*, the petitioners[8] at the Governorship Election Tribunal[9] filed a petition seeking for declaratory reliefs that would ensure that the announcement of the Respondents as the winner of the Governorship Election in River State[10] was vacated.

After victories for the petitioners at both the Tribunal and the Court of Appeal (C.A.), the Respondents (Appellant at the Supreme Court) contested, at pages 42-43 of the report, the authenticity of data from the electronic card reader device. They argued that the examination alone of the card reader device- without an added examination of the paper-based voters' register- cannot provide proof of accreditation or over-voting for an election [10]. The Court, at page 63, agreed, holding that:

> [T]he Tribunal and the Lower Court were unduly swayed by the INEC [Independent National Electoral Commission] directives on the use of the card readers. As held by this court, the INEC directives, Guidelines and Manual cannot be elevated above the provisions of the Electoral Act *so as to eliminate manual accreditation of voters*. This will remain so *until INEC takes steps to have the necessary amendments made to bring the usage of the Card Reader within the ambit of the substantive Electoral* Act [10].

In essence, the Supreme Court found that the provisions of the e-accreditation rules (i.e., those contained in the electoral guidelines) issued by the INEC conflicted with those of the Electoral Act and hence declared that the former were inapplicable.

However, this decision was made at the time when the relevant provisions of the Electoral Act[11] provided only for the manual accreditation of voters. In other words, while the electronic card readers introduced by INEC were applauded by the National Assembly, the same body failed to amend the Electoral Act to legalize said "card readers" as the ultimate determinant of voters' accreditation, thus creating a legislative faux pas[12] with respect to Nigerian voting laws.

[8] On appeal at both the Court of Appeal and the Supreme Court of Nigeria, the nomenclature of the petitioners later changed to "Respondents."

[9] Section 285(2) of the 1999 Constitution of Nigeria (as amended) established the "Governorship and Legislative Houses Election Tribunal" and granted it the original jurisdiction to hear challenges to the Governorship and State Legislative Houses elections.

[10] This is one of the 36 States in the Federal Republic of Nigeria.

[11] Electoral Act 2010 (as amended), s 49(1) & (2) ("Any person intending to vote with this voter's card, shall present himself to a Presiding Officer at the Polling Unit… (The Officer) shall, on being satisfied that the name of the person is on the *Register of Voters,* issue him a ballot paper and indicate on the Register that the person has voted.") (emphasis added)

[12] "Faux pas"- "words or behaviour that are a social mistake or not polite" https://dictionary.cambridge.org/de/worterbuch/englisch/faux-pas Accessed 6th December, 2017.

Therefore, the Supreme Court literally interpreted the extant electoral law, which does not grant any overriding importance to e-accreditation nor provides for it to be the decisive factor concerning the authentication of voters' identities. As succinctly expressed by Justice Nweze, this is the only logical approach since:

> [A]ny attempt to invest it (the Card Reader Machine procedure) with such overarching pre-eminence or superiority over the Voters' Register is like converting an auxiliary procedure-into the dominant method procedure – of proof, that is, proof of accreditation. This is a logical impossibility [16].

Against the background of the foregoing, it will be interesting to predict that further constitutional and statutory concerns might still attend the use of these "e-accreditation machines," even upon the successful amendment of the Electoral Act. However, a Justice of the Supreme Court, Rhodes-Vivour J.S.C, appeared not to foresee any danger. In a dictum, at pages 86-87, his Lordship opined that:

> [The Card Reader] was introduced to improve the accreditation process. The card reader does not violate any law. It makes election credible and transparent when it works properly. *It follows naturally that once the National Assembly amends the Electoral Act to provide for card readers, then card readers would be very relevant for nullifying elections* [17].

Despite the above optimism, this researcher would rather prefer to tread with caution, particularly regarding the idea of relying solely on CEEM for the nullification of elections. The growing incidence of electoral fraud and allegations of electronic hacking[13] call for some circumspection, particularly for nascent democracies in developing countries; however, this subject is beyond the scope of the present study.

On a final note, it should be noted that the Nigerian Senate- with subsequent harmonization also done by the House of Representatives- has heeded the advice of the judiciary and amended the Electoral Act to give statutory recognition to card readers and other technological devices for accreditation purposes. The amendment to section 49 of the Electoral Act now "[m]andates presiding officers to use the Smart Card Reader… to record, verify, confirm or authenticate…the number of accredited voters in the polling unit…"[18].

4 The Hearsay Challenge to the Admissibility of CEEM and Standards in Evaluating Admitted Evidence

Pendleton [16] states that the admissibility framework for computer-generated evidence consists of a four-part analytical framework that includes: the consideration of whether the computer evidence has been authenticated; whether it is hearsay or not; whether it is relevant and not unfairly prejudicial; and whether it is not a privileged communication. With respect to CEEM, issues relating particularly to the "hearsay"

[13] The 2016 general election in the USA is a notable example. The crisis of confidence in electronic voting can be seen in the unending allegations of interference levelled against the Russians and has highlighted the need to improve the security of elections.

component have created some problems that are worth reviewing for the Nigerian Courts.

Furthermore, closely related to the hearsay-admissibility challenge of CEEM is that of the weight ascribable to this species of evidence. Therefore, this section will examine the judicial stance on the hearsay effect of CEEM and also proffer standards for ascribing weight thereto.

4.1 Hearsay Challenge

With respect to admissibility, the holy grail of the appellate courts in Nigeria appears to be that such CEEM must be tendered in court by their makers. Without prejudice to other factors for admissibility (i.e., relevance, best evidence, etc.), this singular factor ("rule against hearsay with respect to electronic evidence") has been decisive for the outcomes of most electoral petitions involving electronic card readers.

In the case of *Emmanuel & Others v. Umana & Others* [17], the Supreme Court was asked to determine, among other things, the question of the propriety of the C.A.'s reliance on legally inadmissible documentary evidence, the makers of which did not testify before the Court of first instance- i.e., the Elections Tribunal.

The case arose out of a contested governorship election held in Akwa Ibom State. At the conclusion of the poll, the INEC declared the Respondent (Appellant at both the C.A. and the Supreme Court) the winner, upon which the petitioners (Respondents at both the C.A. and the Supreme Court) petitioned to the Governorship Elections Tribunal for the nullification of the election. The Tribunal nullified the elections in 18 Local Government Areas (LGAs) but upheld them in the remaining 13 LGAs.

On further appeal against this decision to the C.A., the Appellant's fate was worsened. The C.A. ordered the nullification of the results in the entire State, after which the Appellant further appealed to the Supreme Court. The apex Court, at page 64, agreed with the Appellant that the C.A.:

> "[W]as in error...in relying on...all the documents that the Court[CA] relied upon including but not limited to exhibits… 317[Card Reader Report of Accredited Voters]… when the makers thereof did not testify before the Court…"[17].

In its holdings, the Court merely enforced the unambiguous provisions of the Evidence Act that prohibit legally inadmissible hearsay documents, except in the situations provided for in sections 40-54 of the Evidence Act[15]. Therefore, since the maker of the "Card Reader Report" did not testify, the statutory presumptions of regularity in section 146(i.e. Presumption as to genuineness of certified copies of documents made by public officers); section 148(e) (i.e. Presumption as to genuineness of document directed to be kept by a person by any law); and section 168(1) (i.e. Presumptions of regularity in favor of official act) [15] will hold in favor of INEC.

These presumptions will thus validate INEC's proffered figure of 1, 222, 836 manually accredited voters[14] (which was recorded in the paper voters' register), as opposed to the figure of 438, 127 accredited voters (which the Petitioners alleged were

[14] The petitioner had, relying on the card readers, sought the voiding of the election based on the allegation that the figure was over-bloated and that only 438, 127 voters were accredited.

captured by the electronic card reader device and documented in the tendered "Card Reader Report").

Consequently, to rebut the said presumption of regularity, the proponent of a CEEM- which will adversely affect the results declared by INEC- must ensure that its maker testifies and tenders the CEEM and the Certificate required by section 84(4) of the Evidence Act. The maker does not, however, need to be an expert in computer forensics and it suffices if he/she merely occupies a responsible position with respect to the electronic device/computer.

Odukoya [20] has noted that politicians often employ the power of incumbency to perpetuate themselves in governance. In view of the undermining effect such actions has on the independence of the electoral commission and the judiciary, there is the need for the relaxation of the hearsay rule in electoral litigation. Such an exception to the rigidity of the hearsay rule will ensure that potential evidence for an election challenger is available, irrespective of the antics[15] of the ruling-incumbent party.

This initiative has been adopted by some legal systems. For example, hearsay evidence is no longer inadmissible in the United Kingdom (UK) solely upon the ground that it is hearsay; it is now possible for a party wishing to adduce hearsay evidence that does not fall under any recognized exemption to merely give notice of such to the other party [21]. This implies that the law of evidence in the UK rule takes a flexible attitude towards the admissibility of all evidence and focuses more on evaluating the weight to be attached to such admitted evidence. In such jurisdictions, it can be inferred that CEEM will also not be disallowed merely because the maker was not called as a witness.

Also, in Nigeria, the Evidence Act appears to provide a unique way of ameliorating the "rigidness" of the Hearsay Rule through s. 52 of the Act. That section provides that records in electronic form made by public officers[16]- like INEC officials- are admissible as exceptions to the hearsay rule.

Notwithstanding this provision, the Supreme Court has held in the Nyesom's case (see page 56) that section 52 does not exempt public records (e.g., a certified "Card Reader Report") from the hearsay rule [10]. In other words, the community reading of ss. 52, 104, and 105 of the Evidence Act only makes the exception in s 52 apply to "proof of contents" of such public documents and not to the "proof of the truth of those contents."

In summation, the approach adopted by the UK rules of evidence[17] on hearsay challenges is largely commendable because it is preferable for the ordinarily "inadmissible" hearsay CEEM to be admitted and the court left with the tasked of evaluating its probative-ness and weight.

[15] These antics sometimes result in the strange unavailability of key electoral officers whose testimony could have strengthened the case of the petitioners. Furthermore, where the independence of the judiciary is compromised, the issuance of a "Subpoena ad testificandum" on a key witness might serve no practical purpose.

[16] Specifically, records that are regarded as "certified public documents" under Nigerian laws.

[17] The Nigerian rules of evidence can also adopt this initiative, for example, by admitting a "Card Readers' Report" tendered by a non-maker public officer while preserving the courts' right to ascribe relevant weight thereto.

4.2 Standards in Evaluating the Weight of Admitted CEEM

According to the jurisprudence of the Nigerian courts, cross-examination is one of the measures that judges can rely on in deciding the weight ascribable to any piece of evidence, including electronic evidence/CEEM. In a host of cases, the Courts have reiterated this principle. In Emmanuel's case (see page 66), the principle was eloquently stated thus:

> What is more, there is, even, authority for the view that as "cross examination plays a vital role in the truth –searching process of evidence procured by examination-in chief it relates to authenticity or veracity of the witness, a Court of law is entitled not to place probative value on evidence which does not pass the test of cross-examination[17].

In addition to the "test of cross-examination," the Court, in Nyesom's case (see page 56) has also held that it will:

> "… [Have regard] inter alia, to all the circumstances from which any inference can reasonably be drawn to the accuracy or otherwise of the statement [rendered admissible by the Evidence Act]."[10].

Thus, the Court also laid down the "test of circumstantial inference" as another important guide in evaluating the weight to be attached to an admitted CEEM. Therefore, under the present state of the law in Nigeria, the weight ascribable to a CEEM can pale if the testifying witness's veracity (in relation to the CEEM) is punctured. The same result will occur if there are negating circumstances that will erode the accuracy of the CEEM.

Some basic clarifications, however, need to be made on the standard being proposed. To effectively evaluate admitted "hearsay" CEEM based on the theoretical underpinnings of the RPT, the following is noteworthy:

i. The standard does not seek to replace the operation of the rules of evidence and proof, but rather to complement them.

ii. Therefore, the rules of evaluating evidential weight should be capable of being broadly defined to encompass the drawing of circumstantial inference from all species of evidence.[18]

iii. Appellate courts must be empowered by the rules of court to overturn perverse findings of facts by trial courts.

4.2.1 Structural Sub-theory of the RPT and Election Petitions' Claims

Against the background of our discussion in Section 2, the Courts must evaluate the weight ascribable to CEEM in such a way that the facts are accurately determined and the risks of errors fairly distributed between the parties. Considering, however, the presumption of regularity that holds in favor of the Nigerian electoral body and the complexities of electronic evidence, it is important to propose a standard that will treat the testimonies of the petitioners and the respondents fairly.

[18] For example, sec. 34 of the Nigerian Evidence Act makes provision for the role of "circumstantial inference" in evaluating the weight ascribable to any admitted evidence.

It is this purpose that the RPT serves. If applied to this research need, the structural sub-theory of the RPT entails that the parties to an electoral petition are to come up with stories that either buttress their own claim to electoral victory or negate that of their challenger in the said election.

For example, a petitioner relying on Section s.138(1)c of the Electoral Act that "…the respondent was not duly elected by majority of lawful votes cast at the election"[12] might come up with stories/claims like: there was unauthorized human access to the electronic card readers before, during, or after the accreditation; the card readers malfunctioned at any time during the accreditation process, etc.[19] In other words, the reasoning behind such claims, if successfully established, is to leave the courts with only lawful votes, votes that will ensure that the petitioner is clearly decided as the winner of the election.

The fact that the Nigerian National Assembly recently gave legislative backing to the use of e-accreditation [18] makes it tempting to agree with Justice Rhodes-Vivour that card readers might eventually- though arguably- be relevant in nullifying elections and the Nigerian Courts need to be prepared for the type of stories enumerated above.

However, the structural sub-theory of the RPT does not explain how the fact-finder will be assisted in determining the most plausible of the offered stories. This task is left to the "theory of juridical evidence." to which we shall now turn.

4.2.2 Theory of Juridical Evidence and Weight of CEEM in Election Petitions

If applied to the present research need, this sub-theory of the RPT implies that the trial court is expected to evaluate the effect of the proffered CEEM on the parties' claims by utilizing the tripartite evidential tools (oral, physical, and miscellaneous trial observations). While undertaking these tasks, the trial courts and appellate courts are to be guided by the principles of "coherence"[20] and "rationality"[21] to prevent arbitrary judicial decision-making.

Therefore, in proving his case, a petitioner whose witness, for example, has provided a coherent explanation of incidences of malfunctioning "electronic card reader devices" during accreditation deserves equal weight as an expert witness who testified as to the technical details of such malfunctions. In other words, this sub-theory predicts that extant evidentiary rules might not always be sufficient to explain legal evidence and calls for a more inclusive approach.

Using the earlier analogy of a petition founded on not being "elected by majority of lawful votes cast," a judge is expected to bring his intelligence and knowledge to bear

[19] See generally: the USA Case of *Americans for Safe Access* v. *County of Alameda,* 174 Cal.App.4th 1287, 1291 (2009), where the Court listed the relevant documents that must be produced to authenticate the accuracy of votes produced from a DRE voting machine.

[20] Since the RPT requires the party to bring up their own stories, we will only be concerned with how "coherence" helps the fact-finders to determine the best of the offered stories.

[21] According to Prof. Allen, it might be difficult to give "rationality" a fixed definition; however, it entails that the explanation is consistent, uniform, coherent, simple and economic.

in evaluating the substantiating CEEM. In other words, the evidence supportive of the "allegation of unauthorized human access to the electronic card readers" will only be highly weighted if:

i. the tendering witness was coherent and rational in the explanations that he proffered before the trial court;
ii. the evidence passed the test of cross-examination;
iii. the tendered evidence is consistent with the judge's miscellaneous observations during trial and prior background knowledge on the nature of such evidence;
iv. and despite the evidence being ordinarily "hearsay," it complies with (a)-(c) in such a way that it appears circumstantially superior to the corresponding evidence from the opposing party.

5 Recommendations and Policy Implications

The possible transition of developing democracies (like Nigeria) to full-blown e-voting requires some circumspection. Recognizing the tendencies of some of these developing countries to adopt electoral initiatives from advanced democracies hook, line, and sinker, policy formulators for such countries must -at present- be prepared to combine the positive aspect of e-accreditation with that of manual procedures.

To this end, there is a need for the continuous training of judges, electoral staff, and other stakeholders on the peculiarities of e-accreditation. The current training modules utilized by bodies like the BRIDGE[22] magnanimously recognized that there are "…powerful people used to working with laws and getting their own way with government employees…" [22]. However, to address this shortcoming in electoral litigations, the relative plausibility and inclusionary approach to evaluating CEEM-proposed in this paper- needs to be reflected. This will ensure that the need of justice is served to all parties in an electoral litigation.

6 Conclusion

The dynamics of the modern world demand a dynamic legal system. This paper has sought to broaden the discussion of e-accreditation to include a new frontier- an evidential standard for the admissibility and weight of "hearsay" CEEM.

Recognizing the need to ensure that all parties to an electoral litigation have equal access to evidence, the paper has argued for a relaxed attitude towards the interpretation of hearsay conditions. By relying on advances being made in this regard in some advanced democracies, the paper contends that Nigerian courts cannot afford to be unnecessarily bogged down by the formalities of rules of evidence and procedure. If laws are made for men and not vice versa, then our attitude to admissible hearsay

[22] BRIDGE is an acronym for "Building Resources in Democracy, Governance and Elections".

CEEM, particularly in the context of developing democracies, requires urgent reforms.

It is hoped that the proposed standard will be inclusive and fair to all the parties in a post-election litigation. If this happens, then a culture of trust in electronic accreditation will be engendered and electronic accreditation machines will foster strong democratic institutions.

Acknowledgements

The author is grateful to the DAAD for providing the funding with which this research work was carried out. Special appreciation also goes to Prof. Dr. Tim W. Dornis and Prof. Dr. Axel Halfmeier of the Leuphana Law School for their valuable advice on its original draft. For finding time to shepherd the paper to completion, the author is also indebted to Prof. Jordi Barrat. However, any error(s) discovered in the paper remains that of the author.

References

1. Norris, P.: Why Electoral Integrity Matters. Cambridge University Press, Cambridge (2014).
2. Caltech/MIT Voting Technology Project, 'Voting - What Is, What Could Be (2001)', https://vote.caltech.edu/reports/1, last accessed: 2018/07/01.
3. Maurer, A.D.: E-Voting: What Do Judges Say? In: Maurer, A.D., Barrat, J. (Eds.) E-Voting Case Law: A Comparative Analysis. Routledge (2016).
4. Saphire, R.B., Moke, P.: Litigating Bush v. Gore in the states: Dual voting systems and the fourteenth amendment. 51 VILL. L., 229, 232-233 (2006).
5. Schwartz, P.M.: Voting Technology and Democracy. 77 N.Y.U. L.Q. Rev., 625, 631-640 (2002).
6. Allen, R.J.: Factual Ambiguity and a Theory of Evidence. 88 Northwestern U. Law Rev., 604, 604-634 (1993-1994).
7. Pardo, M.S.: The Nature and Purpose of Evidence Theory. 66 Vanderbilt Law Review, 547, 556(2013).
8. Moore, J.L.: Time for an Upgrade: Amending the Federal Rules of Evidence to Address the Challenges of Electronically Stored Information in Civil Litigation. 50(2) Jurimetrics, 147, 167 (2010).
9. Council of Europe, https://www.coe.int/en/web/electoral-assistance/-/council-of-europe-adopts-new-recommendation-on-standards-for-e-voting, last accessed: 2017/08/01.
10. Nyesom & Others v. Peterside & Others, (2016) LPELR-40036(SC).
11. Federal Ministry of Justice, http://www.justice.gov.ng/index.php/laws/constitution, last accessed: 2018/07/04.
12. ACE Electoral Knowledge Network. Nigeria Electoral Act 2010, http://aceproject.org/ero-en/regions/africa/NG/nigeria-electoral-act-2010/view, last accessed: 2018/07/29.
13. Independent National Electoral Commission: Voter Registration, http://www.inecnigeria.org/?page_id=5198, last accessed: 2018/07/04.
14. Independent National Electoral Commission: Approved Guidelines for 2015 Elections, http://www.inecnigeria.org/?page_id=3463, last accessed: 2018/07/29.
15. Federal Republic of Nigeria National Assembly: Evidence Act, 2011, https://nass.gov.ng/document/download/5945, last accessed: 2018/07/29.
16. Okereke v. Umahi & Others, (2016) LPELR-40035, 37-38 (SC).

17. Emmanuel v. Umana & others, [2016], LPELR-40037 (SC).
18. Policy and Legal Advocacy Center: Factsheet on the Electoral Act Amendment Bill, https://placng.org/wp/category/publications/, last accessed 2018/07/2019.
19. Bench and Bar of Minnesota: Admissibility of Electronic Evidence: A New Evidentiary Frontier, http://mnbenchbar.com/2013/10/admissibility-of-electronic-evidence/, last assessed: 2018/07/29.
20. Independent National Electoral Commission: Conference Papers, http://www.inecnigeria.org/wp-content/uploads/2015/07/Conference-Paper-by-Adelaja-Odukoya.pdf, last assessed: 2018/07/04.
21. Nicoll, C.: Should Computers Be Trusted? Hearsay and Authentication with Special Reference to Electronic Commerce. J. Bus. L. 332, 341 (1999).
22. BRIDGE, http://www.bridge-project.org/en/curriculum/979-modules/1002-electoral-dispute-resolution-synopsis.html/, last accessed: 2018/07/28.

How much does an e-vote cost? Cost Comparison per Vote in Multichannel Elections in Estonia

Robert Krimmer[1] [0000-0002-0873-539X], David Duenas-Cid[1] [0000-0002-0451-4514] Iuliia Krivonosova[1] [0000-0001-7246-1373] Priit Vinkel[2] [0000-0003-0049-1287] and Arne Koitmae[2]

[1] Tallinn University of Technology, Akadeemia tee 3, 12618 Tallinn, Estonia
[2] State Electoral Office of Estonia
robert.krimmer/david.duenas/iuliia.krivonosova@ttu.ee
priit.vinkel/arne.koitmae@valimised.ee

Abstract. We are presenting the results of the CoDE project in this paper, where we investigate the costs per vote of different voting channels in Estonian Local Elections (2017). The elections analyzed involve different processes for casting a vote: Early Voting at County Centers, Advance Voting at County Centers, Advance Voting at Ordinary Voting District Committees, Electronic Voting, Election Day Voting, and Home Voting. Our analysis shows how the administrative costs per e-vote (an electronic vote) are half the price of the second cheapest option (Election Day Voting), representing the most cost-efficient way of organizing elections, given the conditions of this Case Study. Otherwise, different forms of convenience voting have much higher costs, giving us subjects for further discussion on how to organize multichannel elections.

Keywords: Multi-channel Elections, Calculation of Costs, TDABC, BPR.

1 On e-government, e-voting and calculation of costs

Since McLuhan coined the notion of a global village [42] for the current Information Society [56] we adopted, naturalized and routinized the use of technology for several constituents of our daily life. The leap to an online world of Public Administration [36] had often been regarded as a potential cornerstone for managerial reform and creating future systems of governance [45]. In relation to this, e-government, following Yildiz [61] can facilitate better structures for interconnectivity, service delivery [5], efficiency and effectiveness [24, 50], decentralization, transparency and accountability. Citizens, already used to relating with others (friends, family and businesses) use online tools and consider the use of e-government measures as a normal step in the development of technology-based relationships [8].

Estonia is one of the pioneering and leading countries in adopting e-government tools[1, 27, 32, 51], thanks to the three layers forming the backbone of their government services: the X-road system, the electronic ID and the service provision eesti.ee [41]. Amongst the causes for this success Kalvet [27] lists: 1) utilizing an e-commerce role model for the use of ICT in the public sector [55]; 2) the presence of enthusiastic and visionary civil servants who developed information systems in the public sector [63] and politicians focused on developing a program of e-government [17]; 3) a favorable legislative environment towards ICT; 4) stable funding for ICT expenditures; 5) the adoption of the Estonian ID-Card by public administration; and 6) cooperation between the public and private sectors, especially the banking sector *as a generator of expectations regarding e-government services and as a general catalyst for e-government* (p.146). As a result, Estonia represents an ideal venue for observing different dimensions of e-related expressions such as e-government, e-voting, e-banking or e-commerce [30].

1.1 Convenience voting and electoral complexity

The adoption of e-voting strategies can be inserted into the context of the battle against the consolidated tendency for a declining turnout [4, 39], which is challenging global understanding and the functioning of the democratic process. Some of the causes described for understanding this decline have been summarized as 1) the transition to a less competitive electoral scenario, 2) a generational decline in the will to participate in the political process and 3) a transformation of values that lead to political engagement [6]. The disengagement of citizens at elections threatens the correct functioning of democracy by unbalancing the distribution of power and representation between those who participate and those who do not [37], having spillover effects on the global legitimacy of the system of governance and its decision-making [9, 48]. Many governments and Electoral Management Bodies react by actively seeking out, testing and/or implementing improvements to traditional voting systems, presuming that a more convenient voting system will have positive impacts on the turnout at elections [57].

As a result, new systems for early or convenience voting had been proposed in a number of countries [31, 34], and administrative rules and procedures have been adapted to allow citizens to cast their vote at different moments in the election cycle [20], trying to increase the comfort of voters and ease voters' comfort [2, 7]. Administration of elections represents a necessary factor influencing voter turnout: an adequate voting system might not increase the number of voters, but an inadequate one will definitely decrease it. Although election administration differs from context to context, it is still commonplace that new voting channels cannot replace but can only complement existing methods of participation in elections due to the responsibility to provide a service to the entire electorate [19, 62]. However, the opportunity to rethink and optimize electoral administrative procedures when introducing these additional voting channels is often missed.

The Estonian e-vote remote online voting system, in use since 2005, turns Estonia into the only country in Europe (if not in fact the world) to use this without restriction in all types of elections [54]. The Estonian I-voting project was established in order to sustain and increase voter turnout by creating an additional and convenient voting channel that would be in coherence with efficient use of the infrastructure already in existence [28]. Estonian e-voting systems can be considered a successful and widely used voting mode (over 30% in the last three elections) but with an unequal impact in different subpopulations [52].

The adoption of multi-channel electoral systems poses a set of new challenges to be considered by public administrations, including additional workloads for electoral administrations, increased vulnerability from double voting, increased length of voting periods or difficulties derived from overlapping voting periods [59]. Previous research studies to evaluate multichannel elections [34, 60] indicated the three main areas of concern: 1) multiple-channel elections increase the complexity for election administrations; 2) the increase in complexity requires business process reengineering of electoral processes; and 3) it involves analyzing the cost of introducing new voting channels. This situation addresses a different dimension in the debate on elections, how to achieve the desired social goals with a reduced economic impact.

1.2 Cost accounting

The analysis of the costs arising from running elections has attracted researchers' and practitioners' interest, but a large share of the research already conducted on this issue had been focused on the costs for candidates and campaigns [22, 26, 47], the costs for voters [14, 16, 23, 46] or the costs of public information systems [12, 40]. Other projects that addressed the topic revealed 1) the increase in the cost of elections all over the world [44], 2) the need to define different kinds of electoral costs and the analytical scope of the methodology [38], 3) the need to include costs incurred by adding new voting channels, either high one-off costs (e-votes) or transaction costs (postal voting) [35] and 4) the need to overcome the reduced level of transparency and limited opportunities for scrutinizing certain voting modalities [13]. A clear and successfully proven methodology for facing this challenge is still lacking [58], permitting the calculation of costs of multichannel elections overcoming the previous difficulties, amongst others, 1) the lack of depth in approaches for calculating costs based on the assessment of administrative costs through electoral budgets and their division by the number of voters participating [18], the difficulties of uncovering hidden costs and dealing with different accounting systems and governance structures [10, 38] or difficulties relating to the choice of methodology of directly questioning the source (levels of response, overall quality of responses) [25]. Three main problems can summarize the access to the costs of elections: 1) the difficulties in accessing election costs [11], as many democratic governments are not obliged to divulge this information; 2) the difficulties in recovering hidden costs from budgets; and 3) the difficulties of allocating the costs of public infrastructures to the organization of the election.

2 Methodology

For developing the research methodology, we referred to a broader research field of governmental cost accounting and business-oriented methodologies adapted for calculating administrative management costs. Our goal not only relates to detecting potential inefficiencies in the electoral process or to raising awareness of the costs [43], but also, in particular to deliver comparative results of the costs of different voting channels, in order to enrich the existing literature on e-voting and electoral analysis. To achieve this, our proposed methodology relies on the use of 1) Business Process Reengineering (BPR) [3, 21] for facilitating workflow analysis of complex systems (elections); and 2) Activity-Based Costing (ABC) [15, 33] for calculating costs per service/unit produced by the electoral system (votes), in particular, the use of Time-Driven ABC (TDABC) [29], which reduces the volume of data required for conducting the ABC analysis of 1) the practical capacity of resources committed and the costs involved and 2) unit times for performing transactional activities.

Based on this, a model was developed with the following steps:

1. Conducting electoral process modeling based on the analysis of electoral legislation and publicly available internal instructions, complemented with interviews with stakeholders and on-site observations.
2. Creating a list of activities based on findings from Step 1. Select only those activities which are organized differently depending on the voting channel.
3. Identifying resource pools and determining costs assigned to each resource pool.
4. Attributing costs to activities (attribute directly if possible; attribute by proportional time in other cases) in order to receive total cost per activity.
5. Calculating the practical capacity of resources (we set it at 80% of the theoretical full capacity in line with the standard established in accounting research).
6. Dividing total cost per activity by the practical capacity, to receive cost per minute per activity.
7. Dividing time spent on every activity by output to receive cost per output (in our case, per vote or ballot paper) per activity[1]. Multiply this number by the unit cost of a resource pool in order to receive the cost per vote or ballot paper per activity.

[1] In traditional TD ABC the time per item of output is estimated. However, as is the case with elections, we know precisely how much time is spent on every activity, we receive time per item of output in the manner described above.

Total the cost per vote cast for all activities considered, in order to receive the cost per vote used per voting channel.

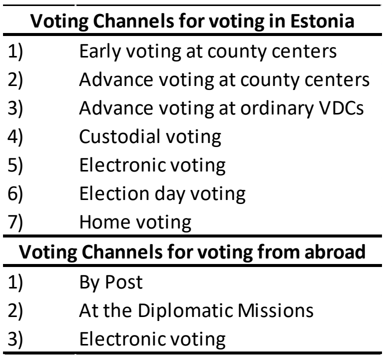
8. Comparing costs per vote cast for different voting channels.

3 Case-study

3.1 Case selection

As was mentioned above, Estonia has a leading position in the development of e-government and I-voting tools, having aroused the interest of many scholars trying to understand the adoption of these tools by citizens [1], its impact on electoral turnout [53] or internal processes in the I-voting system [40], leading many to consider Estonia as a critical case in any relevant research on e-democracy.

Administration of Estonian elections is rather complex, permitting the multichannel analysis proposed. Voters are simultaneously offered multiple voting channels (Fig. 1). However, not all the voting channels are active during every election (voters residing outside Estonia cannot participate in Local Elections) and some of the voting channels, when occurring, overlap both in their periods, like advance voting at county centers, advance voting in ordinary Voting District Committees and Internet voting.



	Voting Channels for voting in Estonia
1)	Early voting at county centers
2)	Advance voting at county centers
3)	Advance voting at ordinary VDCs
4)	Custodial voting
5)	Electronic voting
6)	Election day voting
7)	Home voting
	Voting Channels for voting from abroad
1)	By Post
2)	At the Diplomatic Missions
3)	Electronic voting

Fig. 1. Voting Channels in Estonia.

Two more elements endorse developing a case study in Estonia. Firstly, the fact that Estonian elections by and large use the existing infrastructure, providing an excellent opportunity to test analytical methodologies directed towards delving into hidden costs. Secondly, the involvement of the Estonian Electoral Management Bodies (State Electoral Office) in developing the case study and also the interests of Estonian administration in implementing a similar cost calculation methodology to the one proposed in this research by 2020.

With this background, Estonia has been selected as the first case for us to test our methodology and model. For this analysis, we focus on the most recent elections in Estonia which happened to be the Local Elections taking place in October, 2017.

3.2 Case description

Estonian local elections took place from October 5 -15, 2017, and offered voters seven different voting channels. Overall, it provided a turnout of 586,519 voters (53.3% of the electorate), including 120,888 early and advance voters (20.6% of turnout) and 186,034 e-voters (31.7% of turnout). 279,597 voters cast their votes on Election Day (47.7% of turnout). The results do not represent a big change from previous local elections in terms of overall turnout, following the series of declining turnouts starting in 2009, but indicate a consolidation of the use of e-voting (31.7% of votes cast) and the popularity of voting in county centers (40% of all early and advance votes were cast in 28 county centers, compared to only 60% of advance votes cast in 549 ordinary polling stations).

In order to conduct the cost analysis, we divided voting channels occurring in relation to time of voting:

- *I-voting* (10th to 4th day before Election Day).
- *Early Voting* (10th to 7th day before Election Day).
- *Advance Voting* (6th to 4th day before Election Day).
- *Election Day Voting*.

In relation to the voting location, we consider:

- *Supermarket Voting* - Voting organized in county centers (Early, Advance and Election Day Voting).
- *VDC Voting* - Voting organized in ordinary Polling Stations – Voting District Committees (VDC) according to the Estonian legal system (Advance and Election Day Voting).
- *I-voting*.

This division is based on the following criteria: 1) The differentiation between voting organized online and voting at physical locations (Early, Advance and Election Day Voting) is due to the obvious organizational differences and, as a result, activities and costs involved; 2) voting organized in county centers and voting organized in ordinary VDCs are analyzed separately due to a significant difference in the number of locations (28 county centers compared to 549 ordinary VDCs), staff involved (3-6 members of staff per ordinary VDC and, at least 8 officers per county center), and voting channels offered in these locations (Early Voting is only organized in county centers). Home voting is considered as a subtype of Election Day Voting and, as a result, it is included in this category of our analysis. To analyze it separately, further observations would be required to accurately establish travel time and average number of voters per polling station.

Early voting in county centers is a relatively new voting innovation in Estonia, and it implies that for four days from the 10th to 7th day before Election Day, voters could vote at any of the county centers regardless of the voting district of their residence. In 2017 local elections, 28 county centers were open throughout the country. Half of them were situated in shopping malls, expecting a significant increase in turnout by making the voting process more convenient.

Another important feature of Estonian elections is that early, advance and e-voters are not permitted to override their votes on Election Day. The principle of the precedence of ballot paper voting allows e-voters to override their e-vote with a paper vote but only during the period of early and advance voting, not on Election Day.

3.3 Time frame, processes and activities

As our focus is on cost variation between the different electoral channels present in the Estonian electoral system, we considered the processes occurring in one particular period of the election cycle: the election period [34] (Fig. 2). In Estonia the election period starts 90 days before Election Day with "Informing EU citizens of their right to vote" and finishes three days after the Election Day with the "Resolution of complaints on electoral management". The activities and processes occurring before and after the election period would not add differences to the costs analyzed amongst voting channels, as the activities occurring are the same for every channel

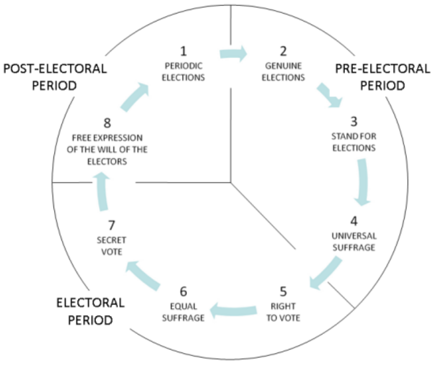


Fig. 2. The Electoral Cycle [34].

Based on the analysis of electoral legislation and publicly available internal instructions, complemented by interviews with municipal secretaries responsible for organizing elections, members of EMBs, members of the National Electoral Committee, and the I-voting auditor, as well as multiple on-site observations across the country, we mapped the electoral processes occurring in the time frame under consideration. Overall, we identified 31 processes with 177 activities among which we selected only major processes which are organized differently, depending on a voting channel which constitutes the third step of our analysis. These processes are as follows:

1. Organization of the voting place.
2. Voter identification.
3. Processing votes.
4. Counting votes.

These four major processes consist of different sets of activities depending on voting channel and voting location. There are 22 activities for I-voting, 8 activities for early and advance voting, and 7 activities for election day voting, all of which will be described in more detail in the following section. This represents our list of activities for TD ABC analysis. During the third step of analysis, we identified the following resource pools: labor costs, printing costs, stationery costs, transportation costs, rental costs, costs of equipment and depreciation costs. We assigned costs to those pools based on electoral budgets available, information derived from procurement contracts, interviews, observations and estimates. In order to assign costs we also considered: the ratio of activities consumed by different voting channels to avoid double counting; the number of times an activity is repeated during the electoral period; the time spent in conducting a certain activity; the number of people participating in a certain activity; and the final number of votes cast through every voting channel. For calculating time, we derived data from log files, on-site observations, legislative regulation and interviews. For the fourth step, labor and transportation costs were attributed directly to activities; other costs were attributed based on the proportion of time every activity consumes. Finally, the steps from the fifth to eighth step were calculated according to the model.

3.4 Description of processes and activities analyzed

Organization of the voting place for Election Day Voting consists of many activities from the delivery of ballots, ballot boxes and other equipment to putting the seal on all paper ballots allocated to a polling station. Moreover, the organization of voting places for Advance Voting requires additional equipment and particular skills from the staff. For Electronic Voting, setting up the voting place is no less complicated. For an e-voter, the voting place is the voting application through which a voter casts a vote. However, the supporting infrastructure without which e-votes could not be cast includes: an electronic ballot box (which is a vote storage server), vote forwarding server and the log server [49].

The process of voter identification differs significantly for the different voting channels. During the Election Day, voter identification occurs based only on the printed voters list. During Advance Voting, those polling places allowing voters from outside their place of residence (county centers) conduct voter identification with the help of the electronic voter registers which are updated daily. Therefore, such voting locations must have computers with access to an updated electronic voter register. For voter identification in I-voting, the voter identifies himself/herself with an ID card used via a card-reader in the voter application. Based on the information retrieved from an ID card, the voter application gives a voter an appropriate list of candidates. To cast a vote, a voter puts a digital signature onto the ballot. Alternatively, identification may be completed with the help of digi-ID or mobile-ID.

Processing votes is the least complicated activity for Election Day Voting, as all votes are stored in ballot boxes, and no additional steps are required before the count. Otherwise, processing votes cast during Advance Voting requires transportation of votes from outside the Voting District (VD) to the appropriate VD/County/National

Electoral Commission. For this purpose, votes should first be sorted according to their VD. This process also requires delivering votes belonging to this VD. Processing e-votes takes place with the help of an electronic ballot box. All other activities associated with it such as removing the information on a voter from a vote take place during the counting process.

Counting votes depends on the format of votes cast: manual counting of paper votes and automated counting of e-votes. All paper votes in Estonia are counted manually, at least two times. No equipment such as scanners is used in the counting process. However, counting advance votes and election day votes also differ from one another. To count advance votes, first votes should be removed from their envelopes. Then, the second stamp should be stamped on every ballot paper. Finally, advance votes are mixed with election day votes and counted together.

Now, when the differences in how four major processes are organized for every voting channel are explained, we move to the description of different sets of activities constituting those processes for every voting channel.

Regarding **Internet-voting**, we consider such activities as: auditing the I-voting system; organizing seminars and training sessions for observers, the media and all those interested in I-voting (activities aimed at building trust); conducting the penetration test of the I-voting system; monitoring the network; activities concerning harmonization between I-voting and paper-based voting (printing and transportation of e-voters' lists, manual transfer of e-voters into printed voter lists); counting and re-counting of votes (these processes are automated, but by law require certain numbers of officers to be present); storage and destruction of e-votes, voter ID cards, and hard drives. Hence, calculating I-voting costs also considers such **cost pools** as transportation and printing costs, alongside labor costs and depreciation costs which take into consideration the expected life span, initial costs of I-voting system acquisition and the cost of updates and replacement.

Regarding **voting organized in ordinary polling stations**, we consider the following activities: delivery of equipment before voting starts (voting booths, ballot boxes, stamps and others); setting up a voting place (installing voting booths, setting up signs giving directions, setting up tables for voting district committee officers); stamping ballot papers before voting (as in Estonia, every ballot must have a stamp from the voting district where it would be issued to a voter); voter identification during voting days; counting ballot papers; transportation of ballot papers for recounting; recounting. Therefore, among the **cost pools** we consider labor costs, transportation costs, printing costs, stationery costs, rental costs for equipment (mainly renting printers and laptops which polling stations need for advance voting and election day voting, but also rental of voting booths as according to our estimation based on interviews and observation, around 25% of VDCs must hire voting booths for elections as they do not possess their own ones).

Regarding **voting organized in county centers**, we consider all the same activities as for voting organized in ordinary polling stations, with one additional activity, which is processing of advance votes from outside the voting district: two members of staff for every county center are obliged to transport votes from outside their voting district to the National Electoral Commission, then, collect home votes, and transport them back to their county. That is how the exchange of votes from outside cast during the advance voting period is currently organized. Another thing to consider is that counting advance votes always requires more resources than counting election day votes, even when it occurs in the same voting settings, because it requires the additional activities which are removing ballots from envelopes and putting a stamp of an appropriate VDC onto a ballot paper for votes cast. In our model, we take this into consideration. Regarding **cost pools**, we consider labor costs, transportation costs, printing costs, stationery costs, and equipment rental costs. Early voting in county centers requires allocating additional voting booths, ballot boxes, envelopes, laptops and printers for those who decide to vote in a different voting place than their own. Such voting places should also have printed lists of candidates available on request for all voting districts. Such voting districts should also have at least part of their staff trained and able to operate laptops with electronic voter registers and printers.

4 Results and costs

The use of TDABC analysis allowed us:

- to consider the different pools of administrative costs incurred during the management of local elections in Estonia, including a) wages, b) depreciation, c) transportation, d) rental, e) printing and f) stationery costs;
- to track the electoral expenses incurred by the different protagonists involved in managing elections, including a) Local Municipalities, b) State Electoral Office, c) Estonian Information System Authority (RIA) and d) others;
- and to allocate those costs to the voting channels, a) Early Voting at County Centers, b) Advance Voting at County Centers, c) Advance Voting at Ordinary VDCs, d) Electronic Voting, e) Election Day Voting (including Home Voting) at County Centers and f) Election Day Voting (including Home Voting) at Ordinary VDCs.

Through process modeling (BPR) we could understand the internal steps for every voting channel and estimate the unused capacity for every model (see Fig. 3). As a result, the TDABC analysis of existing voting channels allows us to allocate numbers to some aprioristic ideas regarding how the costs rise or decline. In particular, the combination of a reduction of use for certain voting channels due to a decline in its popularity but deployment of the same structures and resources (workforce, number of polling stations and working hours), leads to an increase in cost per vote. In particular, our data permits stating that certain forms of Advance Voting have large amounts of unused capacities resulting in low cost-efficiency (higher cost per vote cast) compared to other voting channels.

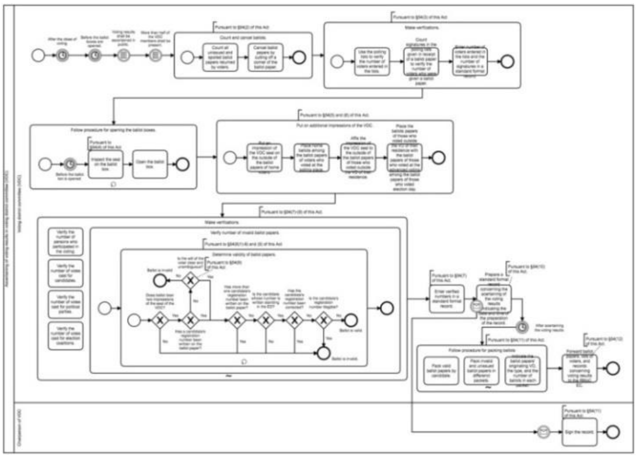

Count and cancel ballots.
Make verifications.
Follow procedure for opening the ballot boxes.
Open the ballot box.
Put on additional impressions of the VDC.
Make verifications.
Verify number of invalid ballot papers.
Determine validity of ballot papers.

Fig. 3. Model of the activity "Ascertaining voting results in a Voting District Committee".

The analysis conducted shows (Fig. 4) that for the Local Elections in Estonia (2017), the most expensive voting channel was Advance Voting in Ordinary VDCs (3) for which the costs considered constituted 20.40 euros per ballot paper. Next comes Advance Voting in County Centers (2) with 6.24 euros per ballot paper and Early Voting in County Centers (1) with 5.07 euros per ballot paper. Regarding Election Day Voting (6), the costs considered constitute around 4.50 euros per vote cast with almost no difference between county centers and ordinary VDCs. I-voting (5) represents the cheapest option carried out in the 2017 Estonian elections, with 2.30 euros per e-vote cast.

Voting Channel	Cost per ballot (in Euro)
Early Voting in country centres	5,07
Advance Voting in country centres	6,24
Election Day Voting in country centres	4,61
Advance Voting in VDC	20,41
Election Day Voting in VDC	4,37
I-Voting	2,32

Fig. 4. Costs for the different voting channels for Estonian Local Elections (2017).

5 Discussion and conclusions

This research has a double and complementary goal to take one step forward in the approach to costs involved for elections. First of all, we aim to use an innovative method in order to count the costs of voting systems to be used in multichannel elections, proving the suitability of its use. Secondly, we aim to put our method into practice in a real electoral context, promoting reflection of the costs of different voting channels and their efficiency.

Regarding the methodological dimension, the methodology we proposed could and should be used in different case studies, should be adapted to the context, or in further elections in the Estonian context, in order to allow more general conclusions to be reached. Accordingly, the results we obtained are valid for the case study we analyzed (Local Elections in Estonia, 2017).

The proposed methodology allowed us to assess with greater accuracy the administrative costs of running elections. The definition of direct and indirect costs incurred by the different protagonists that occur in the organization and development of elections gives a more realistic view of electoral costs, improving previous approaches based on assessing costs by adding up shares of total costs collected from electoral budgets. Secondly, the TDABC methodology allows a more accurate allocation of costs of voting channels, revealing the activities with the heaviest drain on resources that trigger the cost expenditure, facilitating further reflection in the drive for efficiency.

Finally, the use of observation as the main strategy for collecting data allows us to surpass some traditional limitations of calculating electoral costs. Amongst other things, previous researches pointed out the limited access to data on electoral costs and the lack of ability to track expenses as the main constraints for a better fit for analyses. Moreover, this observational approach allows replicating research in other contexts where the availability of information on electoral costs is poor but observation of the electoral process is allowed. In order to test the assumptions derived from our observations, the approach was complemented by a significant number of interviews with polling officers and staff, members of local electoral councils, National Electoral Commission, State Electoral Office of Estonia and other agencies involved in elections. The support of Electoral Management Bodies when providing information and experience-based opinions improves the validity and credibility of the results.

Regarding the cost analysis, we can raise some general statements regarding the Estonian Local Elections (2017): 1) E-voting is the cheapest voting channel proposed in the electoral context analyzed due to the tool's acceptation by citizens and reduced costs involved in deployment. The cost per e-vote cast is half the cost of the second cheapest option; 2) Election Day Voting represents the second cheapest option per vote due to the fact that it is a frequently used voting channel and even with the increased amount of resources deployed; 3) Early and Advance Voting channels are more expensive due to the length of deployment and the lower number of participants that use these channels by comparison; 4) Advance Voting in Ordinary VCD is by far the most expensive channel, at around 18.00 euros per vote more expensive than the cheapest voting channel.

Costs per vote are correlated with resources invested and the popularity of the voting channel. In the search for convenience for voters, e-voting seems to be a good bet in terms of efficiency and success amongst voters, refocusing the debate on suitability to other dimensions (trust, security). The consolidation and success of e-voting in the Estonian electoral context, and its consequent cost efficiency clearly contrasts with other voting channels that consume more resources without achieving such high levels of success. Even so, we would like to stress that the results presented are valid for the elections analyzed, and that a change of voters' electoral behavior in further elections could impact on the distribution of costs by changing them substantially. To better understand electoral costs, this research should be repeated in the same electoral context allowing a comparison between elections.

Finally, the use of TDABC methods in this research, and in future research studies, may have practical implications in terms of rethinking the way elections are organized and formulated; consequently, less efficient voting channels try to maintain the conditions to allow voters to cast their votes in a convenient way but have less impact on reducing public expenditure. Multi-channel elections including e-voting, such as the one analyzed, represent a different and complex reality that can challenge the viability of some paper-based voting channels, especially those with higher unused capacities that reduce the efficiency of the tool.

Acknowledgment. We extend sincere thanks to Breck Shuyler who was assisting us with modelling of BMPNs and on-site observation during the electoral process. This article is based upon work supported by the Estonian Research Council grant (PUT 1361 “Internet Voting as Additional Channel for Legally Binding Elections: Challenges to Voting Processes Re-engineering”, 2017-2020).

References

1. Alvarez, M.R. et al.: Internet Voting in Comparative Perspective: The Case of Estonia. PS Polit. Sci. Polit. 42, July, 497–505 (2009).
2. De Araújo, J.: Improving public service delivery: the crossroads between NPM and traditional bureaucracy. Public Adm. 79, 4, 915–932 (2001).
3. Attaran, M.: Exploring the relationship between information technology and business process reengineering. Inf. Manag. 41, 5, 585–596 (2004).
4. Barrat Esteve, J. et al.: Votacions electròniques: una eina de gestió pública per a la millora de la qualitat democràtica i la participació política. Escola d'Administració Pública de Catalunya, Barcelona (2018).
5. Bekkers, V.J.J.M., Zouridis, S.: Electronic Service Delivery in Public Administration: Some Trends and Issues. Int. Rev. Adm. Sci. 65, 2, 183–195 (1999).
6. Blais, A., Rubenson, D.: The Source of Turnout Decline: New Values or New Contexts? Comp. Polit. Stud. 46, 1, 95–117 (2013).
7. Buckley, J.: E-service quality and the public sector. Manag. Serv. Qual. 13, 6, 453–462 (2003).
8. Carter, L., Bélanger, F.: The utilization of e-government services: Citizen trust, innovation and acceptance factors. Inf. Syst. J. 15, 1, 5–25 (2005).
9. Cavanagh, T.: Changes in American voter turnout 1967-1976. Polit. Sci. Q. 96, 1, 53–65 (1981).
10. Chowdhury, A.: Cost of Voting: Estimating the impact of online voting on public finances. , London (2017).
11. Clark, A.A.: The cost of elections: Money well spent? Polit. Insight. 5, 3, 16–

19 (2014).

12. Codagnone, C.: ’Measuring E-Government: Reflections from eGEP Measurement Framework Experience. Eur. Rev. Polit. Technol. 4, 89–106 (2007).
13. Coleman, S.: Elections in the 21st Century: from paper ballot to e-voting. , London (2002).
14. Colomer, J.M.: Benefits and costs of voting. Elect. Stud. 10, 4, 313–325 (1991).
15. Cooper, R., Kaplan, R.S.R.S.: Activity-Based Systems: Measuring the Costs of Resource Usage. Harv. Bus. Rev. 6, 3, 96–103 (1992).
16. Downs, A.: An Economic Theory of Political Action in a Democracy. J. Polit. Econ. 65, 2, 135–150 (1957).
17. Ernsdorff, M., Berbec, A.: Estonia: the short road to e-government and e-democracy. In: Nixon, P. and Koutrakou, V. (eds.) E-Government in Europe: Re-booting the State. p. 228 Routledge, New York (2007).
18. Ernst & Ernst: Election administration Volume III: Costing Elections. Ill, (1979).
19. Grabenwarter, C.: Report on the compatibility of remote voting and electronic voting with the standards of the Council of Europe. , Strasbourg (2004).
20. Gronke, P. et al.: Convenience Voting. Annu. Rev. Polit. Sci. 11, 1, 437–455 (2008).
21. Grover, V. et al.: The Implementation of Business Process Reengineering. J. Manag. Inf. Syst. 12, 1, 109–144 (1995).
22. Harada, M., Smith, D.M.: You have to pay to play: Candidate and party responses to the high cost of elections in Japan. Elect. Stud. 36, 51–64 (2014).
23. Haspel, M., Gibbs Knotts, H.: Location, location, location: Precinct placement and the costs of voting. J. Polit. 67, 2, 560–573 (2005).
24. Heeks, R.: Understanding e-Governance for Development. i-Government Work. Pap. Ser. 20, 2, 1–27 (2001).
25. James, T.S., Jervier, T.: The cost of elections: the effects of public sector austerity on electoral integrity and voter engagement. Public Money Manag. 37, 7, 461–468 (2017).
26. Johnston, R., Pattie, C.: How Much Does a Vote Cost? Incumbency and the Impact of Campaign Spending at English General Elections. J. Elections, Public Opin. Parties. 18, 2, 129–152 (2008).
27. Kalvet, T.: Innovation: a factor explaining e-government success in Estonia. Electron. Gov. an Int. J. 9, 2, 142 (2012).
28. Kalvet, T.: Management of technology: The case of e-voting in Estonia. In: ICCTD 2009 - 2009 International Conference on Computer Technology and Development. pp. 512–515 (2009).
29. Kaplan, R.S., Anderson, S.R.: Time-driven activity-based costing: a simpler and more powerful path to higher profits. Harvard Bus. Sch. Press Books. 82, 266 (2007).
30. Kassen, M.: Open data and e-government – related or competing ecosystems: a paradox of open government and promise of civic engagement in Estonia.

Inf. Technol. Dev. 1–27 (2017).

31. Kersting, N., Baldersheim, H.: Electronic Voting and Democracy. Palgrave - MacMillan, New York (2004).
32. Kitsing, M.: Success without Strategy: E-government Development in Estonia. Policy & Internet. 3, 1, 1–21 (2011).
33. Kont, K.-R.: What do acquisition activities really cost? A case study in Estonian university libraries. Libr. Manag. 36, 6/7, 511–534 (2015).
34. Krimmer, R. et al.: The Development of Remote E-Voting around the World: A Review of Roads and Directions. Lect. Notes Comput. Sci. 4896, 1–15 (2007).
35. Krimmer, R., Wendt, F.: Costs of Electronic Voting: An Overview. , Vienna (2010).
36. Layne, K., Lee, J.: Developing fully functional E-government: A four stage model. Gov. Inf. Q. 18, 2, 122–136 (2001).
37. Lijphart, A.: Unequal Participation : Democracy ' s Unresolved Dilemma. Am. Polit. Sci. Rev. 91, 1, 1–14 (1997).
38. López-Pintor, R., Fisher, J.: Getting to the Core. A Global Survey on the Cost of Registration and Elections. United Nations Development Programme, New York (2006).
39. López Pintor, R., Gratschew, M.: Voter Turnout Since 1945: A Global Report. IDEA Institute for Democracy and Electoral Assistance, Stockholm (2002).
40. Maaten, E., Hall, T.: Improving the Transparency of Remote E-Voting: The Estonian Experience. In: Krimmer, R. and Grimm, R. (eds.) 3rd International Conference on Electronic Voting 2008. pp. 31–45 Gesellschaft für Informatik, Bregenz (2008).
41. Margetts, H., Naumann, A.: Government as a Platform: What can Estonia Show the World? , Oxford (2017).
42. McLuhan, M.: Understanding Media: The extensions of man. (1964).
43. Mitchell, F.: Implementing Management Innovations, Lessons Learned from Activity Based Costing in the US Automobile Industry (Book)., (2002).
44. Montjoy, R.S.: The Changing Nature . and Costs . of Election Administration. Public Adm. Rev. 70, 6, 867–875 (2010).
45. Moon, M.J.: The Evolution of E-Government among Municipalities: Rhetoric or Reality? Public Adm. Rev. 62, 4, 424–433 (2002).
46. Niemi, R.G.: Costs of voting and nonvoting. Public Choice. 27, 1, 115–119 (1976).
47. Petithomme, M.: Second-Order Elections, but also 'Low-Cost' Campaigns? National Parties and Campaign Spending in European Elections: A Comparative Analysis. Perspect. Eur. Polit. Soc. 13, 2, 149–168 (2012).
48. Salisbury, R.: Research on political participation. Am. J. Pol. Sci. 19, 2, 323–341 (1975).
49. Springall, D. et al.: Security Analysis of the Estonian Internet Voting System. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security - CCS '14. pp. 703–715 ACM Press, New York,

New York, USA (2014).

50. Szopiński, T., Staniewski, M.W.: Manifestations of e-government usage in post-communist European countries. Internet Res. 27, 2, 199–210 (2017).

51. Toots, M. et al.: Success in evoting - Success in edemocracy? The Estonian paradox. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 55–66 (2016).

52. Vassil, K. et al.: The diffusion of internet voting. Usage patterns of internet voting in Estonia between 2005 and 2015. Gov. Inf. Q. 33, 3, 453–459 (2016).

53. Vassil, K., Weber, T.: A bottleneck model of e-voting: Why technology fails to boost turnout. New Media Soc. 13, 8, 1336–1354 (2011).

54. Vinkel, P., Krimmer, R.: The how and why to internet voting an attempt to explain e-stonia. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 178–191 (2017).

55. VVAA: Information and Communication Technologies for the Public Service: A Small States Focus. Commonwealth Secretariat, London (2008).

56. Webster, F.: Theories of the information society. Routledge, New York (2007).

57. Wilks-Heeg, S.: Treating Voters as an Afterthought? The Legacies of a Decade of Electoral Modernisation in the United Kingdom. Polit. Q. 80, 1, 101–110 (2009).

58. Xenakis, A., Macintosh, A.: A generic re-engineering methodology for the organized redesign of the electoral process to an e-electoral process. In: Krimmer, R. (ed.) EVOTE’06, 2nd International Workshop on Electronic Voting. pp. 119–130 Gesellschaft für Informatik, Bregenz (2006).

59. Xenakis, A., Macintosh, A.: Levels of difficulty in introducing e-voting. In: Traunmüller, R. (ed.) Electronic Government: Third International Conference, EGOV 2004 Zaragoza, Spain, August 30 - September 3, 2004 Proceedings. pp. 116–121 Springer-Verlag, Berlin (2004).

60. Xenakis, A., Macintosh, A.: Procedural Security Analysis of Electronic Voting. In: Proceedings of the 6th International Conference on Electronic Commerce. pp. 541–546 ACM, New York, NY, USA (2004).

61. Yildiz, M.: E-government research: Reviewing the literature, limitations, and ways forward. Gov. Inf. Q. 24, 3, 646–665 (2007).

62. Code of Good Practices in Electoral Matters: Guidelines and Explanatory Report. , Venice (2002).

63. EU-8: Administrative Capacity in the New Member States: The Limits of Innovation? (2006).

Legal issues of online participation in municipalities and universities in the Federal Republic of Germany

Frank Bätge[1]

[1] University of Applied Sciences for Public Administration and
management of North Rhine-Westphalia. Haidekamp 73, 45886 Gelsenkirchen, Germany
frank.baetge@fhoev.nrw.de

Abstract. Online participation in elections and referendums has many advantages, most notably the more flexible nature of the voting process and the positive effects on voter turnout. In particular, in the self-governing bodies, an electronic instrument of participation is of particular importance for their members, because it corresponds to the idea of self-administration.

State elections and referendums are strictly governed by the principles of generality, immediacy, freedom, equality, secrecy and transparency of the vote. At present it is not possible (yet) to fully comply with all principles in a purely electronic procedure at the same time. Some compromises may be made in the context of a weighing up, if other important constitutional values are thereby strengthened.

The use of online participation in elections and referendums by self-governing bodies, such as municipalities or universities, is considerably freer. These can install regulations on the procedure by their own statutes. At universities, e-voting has been successfully practiced since the turn of the millennium. Municipalities also have the opportunity to use e-voting in voluntary, non-statutory elections and votes.

This paper explains the legal problems arising from the introduction and implementation of online participation in universities and municipalities and would like to propose some possible solutions.

Keywords: legal issues, elections and votes by self-governing bodies, municipalities, universities, electoral law principles.

1 Introduction

In the Federal Republic of Germany, there are some special public corporations of self-administration with the constitutional right of self-government below the state level. These include, above all, the municipalities and universities (Art. 5 Abs. 3 of the Constitution - Grundgesetz), which have the right to regulate their own interests under the existing federal and state laws on their own responsibility. As a result, they have their own powers and rights, enabling them to issue statutes within their area of responsibility [1]. This constitutional mandate is based on the idea that the members of these self-governing institutions, because of their proximity to the subject matter, are best suited to decide what is appropriate and needed for the well-functioning of their public corporation. It is in line with this concept of self-administration that the members should be enabled to participate as widely and actively as possible in the election of representatives, but also in referendums or consultative participation procedures. The German Constitutional Court (Bundesverfassungsgericht) has therefore formulated that “the image of self-government is significantly influenced by the principle of participation” [2].

To improve effective forms of participation, the Information and Computer Technology (ICT) tools can be used in particular. The introduction of ICT is - besides other positive aspects [3] - especially seen as a necessary step in the fight against declining turnout [4]. Among the ICT instruments, Internet voting is essential. It is a form of remote voting that uses the voter´s personal device to the internet to cast a vote [5]. By opening an online form of participation, an increase in participation can be achieved, as more members are reached, who do not regularly use the conventional methods. Voting through the Internet would also be possible as mobile voting. All members benefit from this flexibility; but especially for people with disabilities or with a foreign residence, the participation in elections and referendums is particularly facilitated. Especially with younger people, life is already largely permeated by the computer and they are familiar with electronic communication in many other online-organized areas of daily life [6]. At least for student body elections at universities this has already been explicitly recognized as a legal consideration factor in upper court case law [7].

On the other hand, internet voting faces well-known serious concerns. These can be summarized under the keywords of “insufficient transparency”, “inadequate procedural controls”, “susceptibility to abuse” and the far-reaching impact of even minor technical errors [8]. For political binding elections, this is very problematic. Due to the importance of these elections and the political pressure [9] legal requirements are stringent. The possibilities to carry out experiments during binding political elections are limited. But on the lower level of non-political elections in municipalities and universities the scope for introducing e-voting tools can be wider.

2 Online participation in self-government practice

At the level of public self-administration, the possibilities of electronic voting are of particular interest to universities and municipalities. Certain universities in Germany have introduced an electronic voting option for the election of their self-governing committees by statute [10]. In the state of North Rhine-Westphalia, there is a new draft law that explicitly allows the introduction of online elections by the universities [11]. Students are generally higher involved in the ICT-tools than the general population and most likely better able to use the e-voting [12]. This is an important motivation for the introduction of e-voting by universities and their authorization by the parliament [13].

Citizen participation as an Internet-based participation also takes place at the municipal level, foremost in the area of information or consultancy. This is mainly due to the higher legal barriers for binding political and decision-oriented citizen participation. In these cases, their results have to be implemented because of legal regulations. At the municipal level, these higher standards exist for binding elections of the councils and mayors. The legal risk factors described above have the consequence that e-voting in binding political elections in municipalities is not allowed at the present time. This is essentially due to a restrictive decision of the Federal Constitutional Court [14]. Decision subject was the use of an offline voting computer for voice input and subsequent automated counting. The Bundesverfassungsgericht deemed this unconstitutional. The principles of publicity and transparency were violated by the use of the voting computer. The public nature of elections is an example of an additional (unwritten) principle which exists in Germany [15].

However, political initiatives are recognizable to create a customer-oriented right to vote, applying the conditions of the Federal Constitutional Court, who would like to increase the turnout by offering more voting options to eligible citizens [16]. The results of an actual study [17] confirm the desire of the citizen. It shows that 56 percent of Germans would like to vote online in the 2017 federal elections. The project fits into the political coalition agreement of the governing parties CDU/SPD/CSU to further develop the digital participation of citizens [18]. The planned introduction of supplementary online elements in the social insurance elections is a further example of this [19]. These efforts are related to the e-democracy initiative developed within the framework of the e-Government project, which intends to improve the political participation of the population on the basis of representative democracy [20]. The introduction of electronic elections is not primarily pursuing rationalization effects of the administration, but should bring a benefit as the proponents hope for an increase in turnout [21].

Unlike the binding political elections and referendums at the state level, municipalities have a greater scope for introducing e-voting in their voluntary participatory projects. This is a consequence of their constitutionally guaranteed self-government in Art. 28 Abs. 2 Grundgesetz [22].

Examples of voluntary participatory projects can be listed [23]:

- The possibility expressly granted in the e-government law of the state of North Rhine-Westphalia for the municipalities to open an electronic participation (§ 18 E-Governmentgesetz North Rhine-Westphalia). In this case the results of the participation are to be announced.
- The legal possibility to organize consultative participation processes (see the Municipal Laws of the Federal States Brandenburg, Lower Saxony and Saarland)
- A local self-commitment through a voluntary statute on the participation process in electronic form [24].
- The election of voluntary advisory councils and advocacy groups such as senior citizens, disabled persons or youth councils by statute (§ 27a Gemeindeordnung North Rhine-Westphalia – Municipal Law).

3 Legal criteria in the introduction and design of online participation

The binding elections of public self-governing bodies in Germany are required by law. They have to comply with fundamental legal principles that are already regulated by the constitution. These include observance of the state structure principles and the relevant electoral principles. The state structure principles are described in Art. 20 Grundgesetz: Rule of law, democracy, republic. Art. 28 Abs. 1 Grundgesetz contains expressly the electoral principles for the election of municipal representations: generality (universality), immediacy, freedom, equality and secrecy of choice. According to the case law of the Federal Constitutional Court [25], the principle of public and transparent voting must be observed. It ensures the regularity and comprehensibility of the electoral processes. This is an essential condition for justified citizens' trust in the correct course of the election.

Unlike the municipal elections, there are no explicit constitutional rules for university elections. But for them essentially the same principles apply. As a result of the self-government guarantee of the universities, the fundamental right of scientific freedom and the principle of democracy in the Constitution, also the electoral law principles apply [26]. Only the principle of equality of choice is modified because of the constitutionally position of the professors in the university committees [27]. This is a result of the constitutional guarantee of scientific freedom in Art. 5 Abs. 3 Grundgesetz.

As a Member State of the Council of Europe, Germany must also comply with the regulations of this international organization. The Council of Europe has elaborated standards offering guidance to the member states on how to regulate the use of e-voting. The Recommendation “Cm / Rec (2017) 5 [1]” of the Committee of Ministers to member States and the detailed guidelines for the implementation of the provisions of Rec 2017 (5) must be considered. The Recommendation extends the definition of e-voting and enlists 49 standards in order to the principles and conditions for democratic elections of the European electoral heritage [28].

Thus, a public election of self-governing bodies under the level of binding political elections is therefore subject to legal conditions. The introduction of e-voting is an optional option, but it must be in line with these fundamental legal principles. The details can be regulated by the self-governing bodies themselves.

In the following, the essential constitutional and legal requirements are presented, which the public self-governing forms have to consider when introducing and designing electronic participation elements. These apply primarily to binding decisions, i.e. those that have to be implemented by law. For purely consultative and legally non-binding online participation, legal limits do not apply directly. A similar reference to such criteria but can contribute to an increase in the acceptance of the results of the participation gained [29].

3.1 Introduction of e-voting

A constitutional principle in democracy is that essential decisions in fundamental areas must not be made by the executive, but by the legislature [30]. Art. 20 Abs. 2 S. 1 Grundgesetz requires that all state power emanate from the people and that the essential decisions of the common good must be made by the directly legislated parliament. This could include the introduction and design of an electronic voting option. The Federal Constitutional Court states in his decision for the use of electronic voting machines in the federal election, that regulations on their use are reserved for parliamentary decision [31]. In the election to the German Parliament (Bundestag), decisions on the admissibility of the use of voting machines and the basic requirements for their use also belong to this.

In elections of self-governing public corporations, the statutory autonomy [32] has an important effect. It may therefore be sufficient for the parliamentary legislator to only make the fundamental provisions himself, and for the rest to grant the self-governing statute autonomy with regard to the more detailed provisions on election and election procedure [33]. Only the most important basic conditions are reserved for the parliamentary decision. These in particular are the electoral system, electoral eligibility, electoral proposals, the electoral register and electoral nominations [34]. If the parliamentary legislator has made such regulations himself, the self-governing could determine the more detailed provisions within its autonomy.

3.2 Compliance with the electoral principles

Considering this background of the described necessity of a legal regulation it has become clear that there are different legal requirements for the various elections. The constitutional requirements for parliamentary elections tend to be higher than those for political elections in municipalities, while elections to other self-governing bodies (for example in universities) have the widest scope for regulation [35].

The particularly stringent requirements for parliamentary elections are linked to the importance of the elected committee as the central legislative organ (i.e. the parliament). The electoral law principles of general, direct, free, equal and secret choice are regulated by the Constitution (Art. 38 Abs. 1 S. 1 Grundgesetz). For elections of the parliaments in the federal states (Landtage), the Constitution (Art. 28 Abs. 1 S. 2 Grundgesetz) also requires a full commitment to the electoral principles.

Although not legislative, but executive bodies (council and mayor) are elected in the local elections of the municipal representations, there is also an explicit constitutional requirement under the Basic Law (Art. 28 Abs. 1 S. 2 Grundgesetz) that the principles apply in their entirety.

In the elections to committees of other public self-governing forms (for example universities and social insurance) expressly written requirements of the Constitution do not exist. The same goes for the online participation below the level of legally standardized elections and referendums (for example citizen surveys, consultations, elections of Seniors Advisory Councils). In these areas, there is more room for maneuver within the self-governing right. Of course, such public committees must also be democratically legitimized. For these reasons, their elections are subject to the constitutional requirements of democracy, the rule of law and the principles of electoral. However, according to the case law of the Federal Constitutional Court, the electoral law principles may still be further restricted with regard to the specific features and specific tasks of the self-governing form [36]. But the Recommendation Cm / Rec (2017) 5 [1] and the guidelines for the implementation of the commissions of Rec 2017 (5) should also be considered. According to Art. 25 Grundgesetz, the general rules of international law are part of federal law. They take precedence over the simple federal laws.

In summary, the electoral law principles have a special relevance for all elections of public committees, but depending on the type of election, this requires a differentiated approach. The strictest requirements can be found in regulated parliamentary and political binding municipal elections, while in the elections of other self-governing bodies the regulator is given more leeway. For the self-governing public corporations, the details of the electoral process, including the modalities of electronic voting, can be determined by them within the framework of the statute of autonomy. The self-governing corporation has to comply with the higher-ranking state laws in substantive law when issuing its regulations.

Essential constitutional minimum requirements for the introduction und use of e-voting in self-governing corporations can be found in the electoral law principles. In democratic elections to all public corporations in Germany, these have to be observed in principle with the specific features of self-governing bodies as shown. Technical issues are primarily related to the security of the systems used. Specifically, it is about compliance with the electoral principles of the generality (ensuring eligibility and authentic voting), the immediacy ("voters and not the computer must make the voting decision"), secrecy (protection against recognizability of voting), and publicity (traceability of their own votes) as well as the risk of manipulation of choice (free and equal suffrage) [37].

For legally binding citizens' decisions in municipalities, the electoral law principles apply with partial restrictions, for example, the local government may comment upon the question put to the vote and even take an own position, which it is denied to them in elections (see the Municipal Laws of the Federal States such as § 26 Gemeindeordnung of North Rhine-Westphalia).

In the following it will be shown that the introduction of electronic voting in elections of self-governing bodies can intervene in the electoral law principles and in state structure principles. But not every intervention leads to a breach of these principles and thus to the illegality of the electronic election. It requires a case-by-case assessment with the advantages of electronic voting, insofar as these include constitutionally relevant values [38]. This balance is reserved for the self-governing corporations of the respective supreme representative body, i.e. the senate of a university or the council of a municipality.

Generality of the vote. First, the introduction of electronic voting could interfere with the electoral principle of universal suffrage. It contains eligibility to vote and eligibility to be elected and ensures equal access to the election [39]. The unjustified exclusion of individual citizens from participation, especially for political, economic or social reasons, is incompatible with this. The electoral participation cannot be made dependent on certain conditions that cannot be fulfilled by everyone. The elections are only universal if there is equality in active and passive suffrage. Therefore, the generality of the election is also a special feature of the principle of equality [40]. In principle, the introduction of electronic voting promotes the principle of generality of choice, provided that it is designed as a further participation option without excluding other participation options. The introduction of electronic voting is currently seen as a tool to increase turnout. However, there is no constitutional obligation for the legislator to introduce this. The Federal Constitutional Court states in a decision on postal vote that the legislature is not obliged by the principle of generality of the vote to create any actual condition for the election, in particular it is not obliged to introduce the postal vote but is entitled to do so [41]. This decision is also transferable to electronic voting, provided that it complies with the constitutional framework in general.

Insofar as it is intended as an alternative means of voting, there are no conflicts with the principle of universal choice. However, it would be problematic to evaluate regulations that provide for an exclusive restriction to the internet choice with one's own computer. In this case, voters who have no computer, no access to the Internet or no computer literacy could in fact be excluded from the right to vote. In particular, elderly persons or those who are not familiar or not sufficiently familiar with the technical possibilities for other reasons would in fact be disenfranchised from the right to vote. People from precarious circumstances who do not have the necessary technical equipment could be equally affected. For these reasons, electronic voting via Internet cannot be the exclusive form. It must be ensured that a vote at the polling station is permitted and that there is a choice of either voting though paper or electronically (off or online) with existing technical equipment (voting computer, screen, online choice and Internet access) and electoral board (also for any inquiries) [42].

Since the principle of generality of choice is also a special feature of the principle of equality, only those persons eligible to vote may therefore participate in the election. In the case of e-voting, a clear authentication of the voter is required.

In the case of the internet choice from the private computer, a corresponding application and an affidavit must be requested for this, which must be provided for security purposes. This is necessary because internet vote, like the absentee ballot, is a distance election [43] and therefore can only be allowed as an exception to the election at the polling station. The "election in secret" must not become the rule in parliamentary elections in the opinion of the Federal Constitutional Court. The constitution in Germany is still based on the model of the polling station election, which makes representative democracy particularly visible. Although e-voting may become the norm in elections to self-governing bodies, it should not be used exclusively [44].

A clear verification of the eligibility to vote by checking an identification code and / or the use of a machine-readable identity card is likewise technically feasible [45].

When voting electronically, it also must be ensured that the operation of the official voting computer (official hardware and software) not only works flawlessly, but also clearly signals, if - as a result of an operating error - an invalid vote should be cast. Otherwise, the person entitled to vote would be unjustifiably excluded from the election. The possibility of correcting the vote before the sending process and the possibility of deliberately invalid voting also must be assured. But these problems are technically manageable [46].

On the other hand, it would be technically more demanding to defend against the risks of general voting in the case of the Internet vote in the event of external attacks on the computer of the person entitled to vote, the transmission of the vote to the electronic ballot box and on the latter itself. This would not only adversely affect the electoral principles of equality and freedom of choice through a manipulative misjudgment of the votes cast. The principle of generality of choice would also be affected if the external attacks prevented altogether the (timely) receipt of the vote cast or manipulated it through the transmission between the computer and the electronic ballot box. Such external risks are conceivable as DoS/dDoS attacks, URL spoofing and Trojan / virus attacks, as well as electronic system failures. In all these cases there is a risk that either the vote is counted as not or not as wanted by the person entitled to vote. Therefore, the technical security standards for the computer used, the transmission path and the electronic ballot box must be at a high standard in order to prevent manipulation or any form of unauthorized interference [47].

Immediacy of the vote. It follows from the principle of immediacy of the election that the appointment of the elected representatives is done by the voter himself and directly. Since the voting person himself must be aware of the effect of his vote, the involvement of independently acting voting computers or other decision-making bodies is excluded. The voter therefore must have the "last word" in the result [48].

In electronic elections of self-governing bodies, the official voting computer and the official election software may only record the vote cast by the voting person. It is technically necessary to ensure that the technology used does not alter the voting decision of the person making the selection by means of independent calculations and thus falsifies it [49]. For the same reason, technical measures against manipulation of the vote by third parties must be fended off. Neither the computer nor a third party may make the voting decision in place of the person voting. Only a properly functioning voting computer, which reduces itself to the electronic capture, storage, and documenting of the vote and is protected against manipulation of third parties, is not a self-referential interim instance and does not affect the principle of the immediacy of the election.

Freedom of vote. Following the principle of free elections, which applies to all elections of self-governing bodies, voters should be able to express their wishes in the election unadulterated. Like all elections, electronic elections must therefore be protected from intervention by the state [50]. In the whole e-voting procedure, it is not allowed, that the office support directly or indirectly influences voters in their voting for a specific electoral proposal (no technical introduction to certain electoral proposals on the electronic referendum, no linkage with campaign pages, no election campaign on the official electoral page, no manipulation of voting) [51].

Equality of the vote. The equal suffrage also demands in elections of self-governing bodies that everyone should be able to exercise his active and passive right to vote in equal manner. Every vote must have the same influence on the election result in the same count and the same score [52]. It is therefore of particular importance in electronic elections to prevent technical manipulation in which a person votes several times or votes of other people are not recorded. This has to be prevented by technical precautions. In particular, it must be ensured in this case that the computer used in electronic voting in the polling station is only activated once per voter for the casting of votes. If the election takes place online, the assigned code must only be valid once. For these cases the possibilities of online or telephone banking, provided that they have proven safe, can be employed. The requirement of equality of choice is followed by the establishment of a checking and supervision process which makes it possible to publicly review the determination of the result of the election. Such requirements on the traceability of the results must therefore be ensured in the electronic election [53].

Secrecy of the vote. The secret suffrage ensures the free choice. This principle requires that the electoral process in elections of self-governing bodies has to be designed in such a way that the persons choosing render their voting decision without third parties being aware of it [54]. The secrecy is already no longer guaranteed if there is the possibility to observe a vote. Protection of the confidentiality requirement cannot be waived by the person voting. In retrospect, it must not be possible to determine how the individual voter has voted. The secret election therefore requires a technical design of the voting process, which makes it impossible to recognize or reconstruct the voting decision of a person choosing [55]. Compliance with the principle of secret ballot is a particular challenge for electronic voting, which requires appropriate technical arrangements. Dangers can occur in the election in the polling station and in the Internet election outside of it. Electronic devices could possibly emanate signals that could be spied out and conclusions could be made about the voting behavior. Even when transmitted over the Internet or other electronic means, the data flows could be monitored. It should also be noted that typing on the computer and "simple mouse clicks", unlike more formal personal signatures in a postal ballot, may reduce the inhibition threshold of counterfeiting [56].

In Internet elections, the detection capabilities are rated even stronger than in offline elections with official voting computer in the polling station. Therefore, special precautions to ensure a secret election or referendum are required. Modern encryption techniques are needed in order to minimize the danger of detection. The signature law with an asymmetric encryption method that is similar to the separation of ballot and ballot papers in the postal vote, could provide a solution. The electoral authority certifies the identity of the voter and continues to ensure the authenticity of his voting decision. For this purpose, a certification authority outside the election organization is deemed necessary [57].

Public and transparent vote. Transparency of the vote, as developed by the Federal Constitutional Court [58], requires that the essential steps in voting and the determination of results can be verified by the voters reliably and without any special expertise [59]. Therefore all major steps of the self-governing body election are subject to public scrutiny. The Federal Constitutional Court, in its important decision on the use of electoral computers, does not postulate the general incompatibility of electoral computers with the principle of publicity. But it is necessary, that the public oversight of the election is strictly adhered [60]. The voters must be able to understand whether the valid votes were correctly assigned to the nominations and that the votes were also correctly determined. On the other hand, it is not enough if a calculation process (internal data processing program) takes place in the voting machine itself, which only IT experts can follow and understand. In that regard, the Federal Constitutional Court expressly demands that even the technical layman must be able to understand the result and how it came about [61].

4 Consideration with legally relevant benefits of online voting

A constitutional justification of the described interventions in the electoral law principles is possible only for other equally important constitutional matters.

In its last decision on the constitutionality of the postal vote, the Federal Constitutional Court once again emphasized that admitting a further voting option can strengthen the generality of choice [62]. This case can be made if the aim is to achieve the widest turnout possible. The principle of generality of the vote is a constitutional demand which might run contrary to the principles of freedom, secrecy and publicity, but is in principle capable of justifying restrictions on other fundamental decisions of the constitution. It therefore remains to be noted that the universal suffrage is an important factor to be considered in favor of electronic voting. But it does not grant any subjective entitlement to the introduction of electronic voting.

In a representative democracy, state power is not executed directly by the people, but by public representatives of the people legitimized by elections. Elections are therefore of high importance for democracy and the legitimacy of the people's representatives. It is for the valid representation of the people necessary to demand a sufficient level of legitimacy. Basically, a low voter turnout does not mean that the democratic legitimacy of a representative body must be questioned. However, minimum limits and developing tendencies must also be taken into account. The Constitutional Court of North Rhine-Westphalia, for example, has obliged the legislator to keep an eye on electoral conditions as to whether the existing electoral system will be able to convey the required democratic legitimacy in the future as well [63]. If the electoral legislature determines that circumstances have changed, it has to be taken into consideration. In elections with a higher turnout, a broader legitimacy of the elected representatives should be expected. This assessment does not lead to a gradation of competences. But a steady decline in voter turnout over unspecified minimums is considered so critical that even this may result in a legislator´s duty to intervene. This certainly does not have to lead to an electronic election. However, such instruments for increasing voter turnout in this context are to be regarded as very relevant factors for consideration.

Other benefits of electronic voting include avoidance of unconsciously mislabeling the ballot by the voter, unintentional counting errors, and misinterpretation of the will to vote. These aspects support the principle of equality of the vote and can therefore, in principle, also be given legal regard as a consideration-relevant advantage [64].

Another factual advantage of electronic voting lies in the faster counting and the associated relief of the electoral boards and possibly related cost savings. From the perspective of the Higher Administrative Court of Thuringia, these aspects of electronic voting can also be considered when making a decision [65].

The legislature only has a narrow margin considering infringements of the electoral law principles. They can only be justified by a legitimate aim important enough to balance the importance of electoral law. Although the legislature has a wide margin of discretion as regards the concrete form of the electoral procedure, the essence of the electoral law principles must not be infringed.

The positive effects on the principles of universal and equal elections and on the principle of democracy are very important in the face of decreasing turnout. The examination of the introduction of e-voting is therefore legitimate, despite the legal obstacles imposed by statutes. For self-governing bodies, the benefits of e-voting are even greater, as their image is significantly influenced by the principle of participation. However, the associated limitations of the electoral law principles are only sustainable if their essence remains untouched.

5 Legal conditions for the introduction of e-voting in self-governing bodies

In the municipalities and universities e-voting can be introduced by statutes in compliance with the aforementioned principles. It is necessary that the following key points in particular should be observed [66]:

- Clear regulations are required as to which safety measures are required and which minimum standards must be met.
- The voting decision must arrive unadulterated in the electronic ballot box and must be counted as such.
- An espionage on the election process must be avoided by appropriate technical precautions. The technical security systems must be defined in the regulations and verified, certified and used consistently in order to prevent manipulation and spying by public authorities.
- It is also necessary to regulate how public enumeration should be carried out.
- Internet dialing from your home computer should not be the only voting option you can choose.
- A neutral official committee has to verify the functionality of the used electronic official hardware and software and their correct representation of the suffrage. This includes not only reviews based on objections and certification, but also random sampling.
- Only electronic software used by the examining and releasing public authority may be used for voting and counting.
- The votes may not be stored exclusively on an electronic memory; a visible paper model or other layman-traceable and functional documentation system must enable a count-in ("inspection of the eye and comprehension" of the voters).
- In order to exclude double votes or a falsified vote, a corresponding identification of the person making the selection is required by the public office.

- It must be possible for the votes cast to be reviewed by the voters before counting.
- Also, in the case of electronic voting, there must be the possibility to cast an invalid vote or no vote.

6 Outlook for further development

Therefore, an interdisciplinary approach and the cooperation of experts for data security, computer science, cryptology, suffrage, etc., is required in order to develop appropriate solutions that ensure the required minimum level of security and traceability. It is advisable to first look at the procedures for electing public bodies outside the parliamentary and political binding elections of the councils and mayors in municipalities. In this area already exists case law, which recognizes a far-reaching statute autonomy and thus makes the introduction of electronic elections easier.

Due to the increasing degree of digitization, the legislator will increasingly have to take into account the legal chances of introducing electronic participation in the future. These include, in particular, the possible positive effects on the participation in participation, the simplification and acceleration of the procedure, as well as the avoidance of unconscious false identifications and counting errors. The Federal Constitutional Court has not closed the door for an electronic vote but formulated specific demands requirements for this purpose to the legislature. In particular, in elections and referendums in institutions of self-administration valuable insights can be gained in this regard, legislators should also be able to gradually make use of statutory experimentation and exemption clauses to enable further forms of electronic participation projects.

At university level, the benefits of online voting are particularly evident given the membership structure [67]. The execution of the electronic election is possible legally secure in this area in compliance with the criteria above. The parliamentary legislature should also use the means at its disposal to further dismantle legal obstacles in this area when formulating the university laws [68].

At the municipal level, voluntary forms of participation can introduce, test and evaluate electronic elements [69]. Even if the detailed structure of the participation procedure is not prescribed by law but is to be regulated voluntarily by statute, a fundamental measure of security and freedom from manipulation must also be guaranteed in these matters for constitutional reasons. For reasons of acceptance, the voluntary electronic participation procedures should therefore not be invalidated or downgraded by insecure procedures. It is therefore advisable to have statutory provisions with sufficiently specific regulations.

References

1. Bundesverfassungsgericht, Urteil vom 21.11.2017 - 2 BvR 2177/16 -, juris, Rn. 75 ff.
2. Bundesverfassungsgericht, Beschluss vom 19.11.2014 - 2 BvL 2/13 -, juris, Rn. 52.
3. OSCE, ODIHR, Handbook for the Observation of New Voting Technologies, p. 5 (2014).

4. Loeber, L., Legislating for e-enabled elections: dilemmas and concerns for the legislator, Proceedings E-Vote-ID, pp. 139, 140, TUT Press (2016).
5. Hill, R., E-Voting und the Law – Issues, Solutions, and a Challenging Question -, Proceedings E-Vote-ID, pp. 123, 128, TUT Press (2016).
6. Danzer, S., Customer-oriented voting rights, KommunalPraxis Wahlen, pp. 2-6 (2015); Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
7. Thüringer Oberverwaltungsgericht, Urteil vom 30.05.2013 - 1 N 240/12 -, KommunalPraxis Wahlen with a comment from Bätge, F., pp. 37-44 (2015).
8. Hill, R., E-Voting und the Law – Issues, Solutions, and a Challenging Question -, Proceedings E-Vote-ID, pp. 123, 131, TUT Press (2016).
9. Loeber, L., Legislating for e-enabled elections: dilemmas and concerns for the legislator, Proceedings E-Vote-ID, pp. 139, 142, TUT Press (2016).
10. Rüttger, M., Online university election - my good law, duz MAGAZIN 09/2016.
11. Landtag NRW (Parliament of North Rhine-Westphalia) Gesetz zur Änderung des Hochschulgesetzes, Referentenentwurf der Landesregierung vom 15.5.2018, Vorlage 17/784, § 13.
12. Loeber, L., Legislating for e-enabled elections: dilemmas and concerns for the legislator, Proceedings E-Vote-ID, pp. 139, 142, TUT Press (2016).
13. Landtag NRW (Parliament of North Rhine-Westfalia) Gesetz zur Änderung des Hochschulgesetzes, Referentenentwurf der Landesregierung vom 15.5.2018, Vorlage 17/784, Amtliche Begründung (official reasoning) zu § 13.
14. Bundesverfassungsgericht, Urteil vom 3.3.2009 – 2 BvC 3/07 -, juris (Wahlcomputer).
15. Maurer, A. D., Updated European Standards for E-voting, Proceedings EVote-ID, pp. 189, 194, TUT Press (2017).
16. Danzer, S., Customer-oriented voting rights, KommunalPraxis Wahlen, pp. 2-6 (2015); Weiler, T. Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015); see to political initiatives: Fahimi, Y. in Frankfurter Allgemeine Zeitung v. 27.12.2014 (Increase turnout through more choice) and Landtag NRW (Parliament of North Rhine-Westphalia), Gesetz zur Änderung des Hochschulgesetzes, Referentenentwurf der Landesregierung vom 15.5.2018, Vorlage 17/784, § 13.
17. Kaspersky, https://www.kaspersky.de/blog/kaspersky-studie-so-steht-deutschland-zum-thema-online-wahlen/13493/ (2017).
18. Koalitionsvertrag CDU/SPD/CSU vom 12.3.2018, pp. 37, 163.
19. Bundestag, Entwurf eines Fünften Gesetzes zur Änderung des Vierten Buches des Sozialgesetzbuchs und anderer Gesetze (5. SGB IV-ÄndG), BT-Drs. 18/3699; see the expert hearing from 2.2.2015: https://www.bundestag.de/dokumente/textarchiv/2015/kw06_pa_arbeit/357030.
20. Danzer, S., Customer-oriented voting rights, KommunalPraxis Wahlen, pp. 2-6 (2015); Weiler, T. Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015); see to political initiatives: Fahimi, Y. in Frankfurter Allgemeine Zeitung v. 27.12.2014 (Increase turnout through more choice) and Landtag NRW (Parliament of North Rhine-Westphalia), Gesetz zur Änderung des Hochschulgesetzes, Referentenentwurf der Landesregierung vom 15.5.2018, Vorlage 17/784, § 13.
21. Weiler, T. Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
22. Bundesverfassungsgericht, Beschluss vom 19.11.2014 - 2 BvL 2/13 -, juris, Rn. 52.
23. Bätge, F./Gerl, K, E-participation in the municipalities in North Rhine-Westphalia, Kommune 21, pp. 44-45 (2018).

24. City of Bonn, Leitlinien der Bürgerbeteiligung (Guidelines of citizen participation), http://www.bonn.de/rat_verwaltung_buergerdienste/buergermitwirkung/index.html?lang=de.
25. Bundesverfassungsgericht, Urteil vom 3.3.2009 – 2 BvC 3/07 -, juris (Wahlcomputer).
26. Bundesverfassungsgericht, Urteil vom 29.5.1973 - 1 BvR 424/71 -, juris (university judgment).
27. Bundesverfassungsgericht, Beschluss vom 9.4.1975 - 1 BvL 6/74 -, juris, Rn. 27.
28. Maurer, A. D., Updated European Standards for E-voting, Proceedings EVote-ID, pp. 189-206, TUT Press (2017).
29. Loeber, L., Legislating for e-enabled elections: dilemmas and concerns for the legislator, Proceedings E-Vote-ID, pp. 139, 142, TUT Press (2016).
30. Bundesverfassungsgericht, Urteil vom 3.5.2016 - 2 BvE 4/14 -, juris, Rn. 52.
31. Bundesverfassungsgericht, Urteil vom 3.3.2009 – 2 BvC 3/07 -, juris (Wahlcomputer).
32. Bundesverfassungsgericht, Beschluss vom 2.11.1981 - 2 BvR 671/81 -, juris, Rn. 13.
33. Bundesverwaltungsgericht, Beschluss vom 14.3.2018 – 6 BN 3/17 -, juris.
34. Thüringer Oberverwaltungsgericht, Urteil vom 30.05.2013 - 1 N 240/12 -, KommunalPraxis Wahlen, pp. 37-44 (2015).
35. Loeber, L., Legislating for e-enabled elections: dilemmas and concerns for the legislator, Proceedings E-Vote-ID, pp. 139, 142, 143, TUT Press (2016).
36. Bundesverfassungsgericht, Urteil vom 29.5.1973 - 1 BvR 424/71 -, juris (university judgment).
37. Bundesverwaltungsgericht, Beschluss vom 14.3.2018 - 6 BN 3/17 -. juris; Thüringer Oberverwaltungsgericht, Urteil vom 30.05.2013 - 1 N 240/12 -, KommunalPraxis Wahlen, pp. 37-44 (2015); Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
38. Thüringer Oberverwaltungsgericht, Urteil vom 30.05.2013 - 1 N 240/12 -, KommunalPraxis Wahlen, pp. 37-44 (2015); Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015); Landtag NRW (Parliament of North Rhine-Westphalia) Gesetz zur Änderung des Hochschulgesetzes, Referentenentwurf der Landesregierung vom 15.5.2018, Vorlage 17/784, Amtliche Begründung (official reasoning) zu § 13.
39. Bundesverfassungsgericht, Urteil vom 13.2.2008 – 2 BvK 1/07 -, BVerfGE 120, p. 82.
40. Bundesverfassungsgericht, Urteil vom 13.2.2008 – 2 BvK 1/07 -, BVerfGE 120, p. 82.
41. Bundesverfassungsgericht, Beschluss vom 9.7.2013 – 2 BvC 7/10 -, KommunalPraxis Wahlen, pp. 77-79 with a comment from Bätge, F. (2013).
42. Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
43. Hill, R., E-Voting und the Law – Issues, Solutions, and a Challenging Question -, Proceedings E-Vote-ID, pp. 123, 128, TUT Press (2016).
44. Recommendation CM/Rec (2017)5[1], Appendix I – E-Votings Standards. Universal suffrage.
45. Danzer, S., Technical developments and their influence on the right to vote, KommunalPraxis Wahlen, pp. 7-14 (2011).
46. Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
47. Danzer, S., Technical developments and their influence on the right to vote, KommunalPraxis Wahlen, pp. 7-14 (2011).
48. Bundesverfassungsgericht, Beschluss vom 26.2.1998 – 2 BvC 28/96, NJW 1998, p. 2892.

49. Recommendation CM/Rec (2017)5[1], Guidelines for the implementation of voting secrecy recommendations, No. 21 and No. 23.
50. Bundesverfassungsgericht, Urteil vom 10.4.1984 – 2 BvC 2/83 -, BVerfGE 66, p. 369.
51. Recommendation CM/Rec (2017)5[1], Guidelines for the implementation of free suffrage recommendations, No. 10.
52. Bundesverfassungsgericht, Urteil vom 13.2.2008 – 2 BvK 1/07 -, BVerfGE 120, p. 82.
53. Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
54. Bundesverfassungsgericht, Beschluss vom 16.7.1998 – 2 BvR 1953/95 -, BVerfGE 99, pp. 1, 13.
55. Recommendation CM/Rec (2017)5[1], Guidelines for the implementation of free suffrage recommendations, No. 10.
56. Danzer, S., Technical developments and their influence on the right to vote, KommunalPraxis Wahlen, pp. 7-14 (2011).
57. Danzer, S., Technical developments and their influence on the right to vote, KommunalPraxis Wahlen, pp. 7-14 (2011); Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
58. Bundesverfassungsgericht, Urteil vom 3.3.2009 – 2 BvC 3/07 -, juris (Wahlcomputer).
59. Recommendation CM/Rec (2017)5[1], Appendix I – E-Votings Standards. Transparency.
60. Bundesverfassungsgericht, Urteil vom 3.3.2009 – 2 BvC 3/07 -, juris.
61. Bundesverfassungsgericht, Urteil vom 3.3.2009 – 2 BvC 3/07 -, juris.
62. Bundesverfassungsgericht, Beschluss vom 9.7.2013 – 2 BvC 7/10 -, KommunalPraxis Wahlen, pp. 77-79 with a comment from Bätge, F. (2013).
63. Verfassungsgerichtshof of North Rhine-Westphalia, Urteil vom 26.5.2009 – 2/09 -, juris.
64. Weiler, T., Can electronic elections be constitutional?, KommunalPraxis Wahlen, pp. 7-19 (2015).
65. Thüringer Oberverwaltungsgericht, Urteil vom 30.05.2013 - 1 N 240/12 -, KommunalPraxis Wahlen, pp. 37-44 (2015).
66. Bundesverwaltungsgericht, Beschluss vom 14.3.2018 - 6 BN 3/17 -. juris; Thüringer Oberverwaltungsgericht, Urteil vom 30.05.2013 - 1 N 240/12 -, KommunalPraxis Wahlen, pp. 37-44 (2015); Rüttger, M., Online university election - my good law, duz MAGAZIN 09/2016.
67. Rüttger, M., Online university election - my good law, duz MAGAZIN 09/2016.
68. Landtag NRW (Parliament of North Rhine-Westphalia), Gesetz zur Änderung des Hochschulgesetzes, Referentenentwurf der Landesregierung vom 15.5.2018, Vorlage 17/784, § 13.
69. Bätge, F./Gerl, K, E-participation in the municipalities in North Rhine-Westphalia, Kommune 21, pp. 44-45 (2018).

Improving Models

Process Models for Universally Verifiable Elections

Rolf Haenni, Eric Dubuis, Reto E. Koenig, and Philipp Locher

Bern University of Applied Sciences, CH-2501 Biel, Switzerland
{rolf.haenni,eric.dubuis,reto.koenig,philipp.locher}@bfh.ch

Abstract. In this paper, we analyze the process of performing the universal verification of an electronic election. We propose a general model of the election process and define the data flow into the verification process. We also define the purpose and outcome of the verification process and propose some general categories of tests to be performed during the verification. As a guideline for dealing with negative verification outcomes, we propose some general evaluation criteria for assessing the impact and consequences of the encountered problem. Finally, we generalize the proposed process models to the case of hybrid elections, in which multiple voting channels are available simultaneously. The primary target audience of this paper are people in charge of implementing and organizing verifiable elections in practice.

1 Introduction

Universal verifiability is a key concept for making electronic voting systems secure enough for using them in real political elections. It is a counter-measure against all sorts of threads from very powerful adversaries, which for example may try manipulate the election result by taking control over some of the central system components. To prevent such attacks, the system generates some public election data during the election process, which can be used to reconstruct the final election result in a publicly verifiable manner. Independent third parties (auditors) can then be invited to verify the correctness of the election result based on the cryptographic evidence included in the public election data. Provided that the verification has succeeded, one can then conclude that no such attacks have been conducted. By providing this simple functionality, universal verifiability is a very important trust-establishing measure. Its ultimate goal is to convince even the losers of an election to accept the result [7, 11].

1.1 Universal Verifiability in Practice

One of the major challenges of building a universally verifiable election system is to provide verifiability simultaneously with vote secrecy. Many cryptographic protocols have been invented for that purpose. Their main problem is to define the verification process in a way that the correct election result can be reconstructed without explicitly decrypting the submitted encrypted votes. For this, some anonymization mechanism must be applied to the submitted votes to unlink them from the voters. Techniques for solving this problem, for example mix-nets or homomorphic tallying, are well-understood today and widely applied. Practical systems using these technologies have been introduced for both academic and real-world purposes [2–6].

Based on today's generally accepted understanding that verifiability is crucial for electronic elections, countries such as Switzerland and Estonia have decided to update the requirements for their existing e-voting systems. The following quote from the *Federal Chancellery Ordinance on Electronic Voting* (VEleS) underlines this change of paradigm in Switzerland [1, page 3]:

> *"Auditors receive proof that the result has been ascertained correctly. They must evaluate the proof in a observable procedure. To do this, they must use technical aids that are independent of and isolated from the rest of the system."*

To fulfill the extended requirements, the two remaining Swiss e-voting system providers have launched corresponding development projects. By releasing a detailed and comprehensive protocol specification together with two different proof-of-concept implementations [8,9], the CHVote project of the State of Geneva has reached an important milestone in 2017. The launch of the new system, which is currently being developed according to the specification, is planned for the 2019 parliament elections. Similar plans exist for the system offered by the Swiss Post AG, which has officially reached an intermediate expansion stage in early 2018. In both projects, the legal ordinance is clear about implementing proper verification processes along with the introduction of the next-generation systems. However, since VEleS does not further specify the details of such processes, it does not provide sufficient legal grounds for most of the conclusions and recommendations contained this paper.

1.2 Goals and Overview

Despite the recent developments in Switzerland and other places in the world, only little experience exists with respect to conducting an actual verification process for real political elections. The foremost problem is the necessity of providing suitable *technical aids* that offer the desired functionality while satisfying the requirement of being independent from the rest of the system. In some of the above-mentioned systems, such technical aids have never been developed. This leads to a paradoxical situations, where systems are promoted as (potentially) verifiable, but without offering the full package for performing an actual verification.

Another problem is the lack of a common understanding of the exact purpose of a verification and the necessary processes around it. Simple questions like what are the exact input data of a verification process and what are the possible verification results have never been defined in a precise manner. Such a high-level view of the verification process is the main topic of this paper. The goal is to lay the foundations for introducing universal verification into existing and future electoral processes. For this, we look at the commonalities of existing e-voting protocols, propose a high-level summary of the relevant data flow, and finally derive general models for both the election and the verification processes. The paper is written mainly from a technical perspective. Related political, legal, or sociological questions are deliberately left aside.

We will start in Section 2 with a general model of the election process, which defines the principal data flow. This model is general enough to be applicable to both electronic and non-electronic election processes. Based on this model, we propose a definition of the verification process. Particular attention is given to the verification result, which we decompose into five main categories. We also discuss the development process of corresponding verification software. In Section 3, we use the generality of the election process model to define corresponding processes for hybrid voting systems, which provides multiple (electronic and non-electronic) voting channels simultaneously. In this particular setting, additional considerations are necessary to guarantee the completeness of the verification chain. Section 4 summarizes the findings and concludes the paper.

2 Universally Verifiable Elections

An election system's principal function is to establish the correct election result based on the votes submitted by the voters. This should be done in a way that even the losers of the election will accept the result as correct. In a paper-based election system, this functionality is achieved by involving trustworthy people from all parties in the tallying process. In case of observed or suspected irregularities, election authorities can order a re-tally of the votes by independent third parties to remove any existing doubts. In an electronic election system, this is exactly the purpose of conducting a universal verification, but the evidence necessary for inferring the correctness of the result is derived from cryptographic methods rather than human supervision. Irregularities caused by attacks or software bugs can then be detected in a reliable way. The purposes of re-tallying paper votes and universally verifying electronic votes are therefore largely equivalent.

2.1 Election Process

In order to define universal verification more precisely, we must first introduce an abstract model of an election process. To provide compatibility with most existing election protocols, we suppress technical details as far as possible. A common denominator is the *election period*, during which voters can submit their votes. Independently of the exact length of this period, it defines a natural decomposition of the whole election process into three consecutive phases:

$$\text{pre-election phase} \quad \Rightarrow \quad \text{election phase} \quad \Rightarrow \quad \text{post-election phase.}$$

Each phase generates its own part of the *election process data*, which contains the auxiliary cryptographic evidence required to perform the verification of the election result. For the general understanding of the election and verification processes, it is not necessary to further specify the exact content of this data, but it is important to keep in mind that this data is public. Usually, it is written to a public bulletin board, from which it can be retrieved by anyone who wants to perform the verification. The election process data is depicted in Figure 1 as one of the main outputs of the election process.

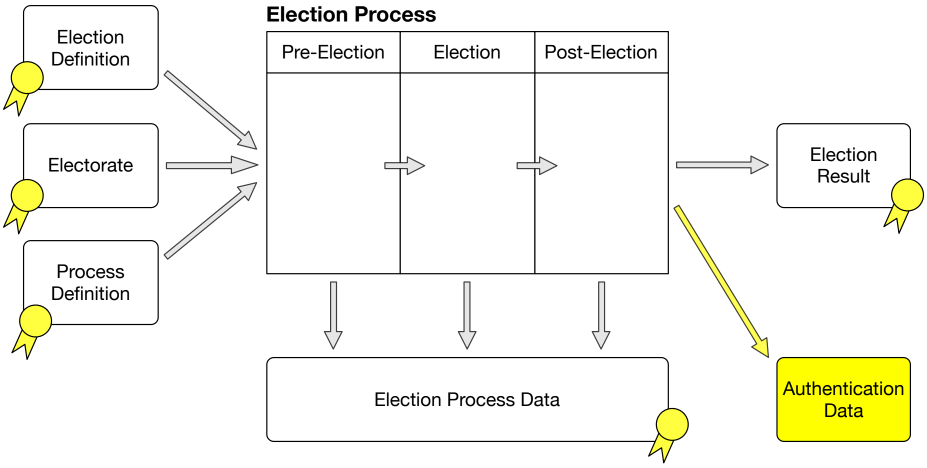


Fig. 1: Abstract model and data flow of an election process.

One of the main inputs of the election process is a document called *election definition*, which defines the details of the election, for example the questions and voting options in a referendum or the list of candidates and election rules in an election. A second input document, which we call *electorate*, contains the list of eligible voters. This document is needed to determine the voter's eligibility and therefore decide about the validity of a submitted vote. Another input document, which we call *process definition*, specifies the details of the election process, for example the start and the end of the election period, but also the identities of the parties and authorities involved in the process or the cryptographic parameters to be used. The party responsible for providing the three input documents is called *election administrator*. Our distinction between election definition and process definition is important in the hybrid setting discussed in Section 3, where a single election definition is combined with two or more process definitions.

The most important output of the election process is the *election result*. We do not further specify the contents of this document, except that we assume that it summarizes the outcome of the election tally, for example by summing up the number of yes/no-votes in a referendum or by simply enumerating all decrypted votes in cleartext. This document represents therefore the official result, which is publicly announced in the aftermath of the election. For this, it is important for this document to contain a signature from the election administrator, which guarantees the correctness of its contents.

The same remark about containing a signature holds for the three input documents and for most parts of the elections process data. We summarize this aspect of the model by assuming an additional output called *authentication data* (yellow-highlighted in Figure 1). In a purely electronic setting, this output will consists of a list of digital signatures with corresponding certificates, from which the authenticity and integrity of all input and output documents can be inferred. As we will see in the next subsection, this aspect defines a particular category of verification steps, which can be performed independently of the rest.

We already mentioned that we kept this process model simple enough for applying it also to the case of non-electronic elections. In that case, some (but not necessarily all) of the involved documents will be paper documents signed by the people that generated them. They may contain declarations that certain manual tasks of the process have been conducted according to the specified procedures. In case of detected irregularities, the existence of such documents can help in identifying the person responsible for causing or overlooking the problem. They are therefore needed for ensuring the plausibility of the election result.

2.2 Verification Process

The process of verifying an electronic election based on the available public data is depicted in Figure 2. The input of the verification process consists of all the public inputs and outputs of the election process model from the previous section. We refer to it as the *election data* and assume that it is available to any person who wants to perform a verification. Note that we consider the election result as part of the election data, which must be checked for correctness. The purpose of the verification process is to perform a series of tests on the election data, which collectively give enough evidence to assess the correctness of the election result. A compilation of the results obtained from performing the necessary tests is what we call the *verification report*. This document is the principal output of the verification process. The software that generates the verification report based on the election data is called *verifier*.

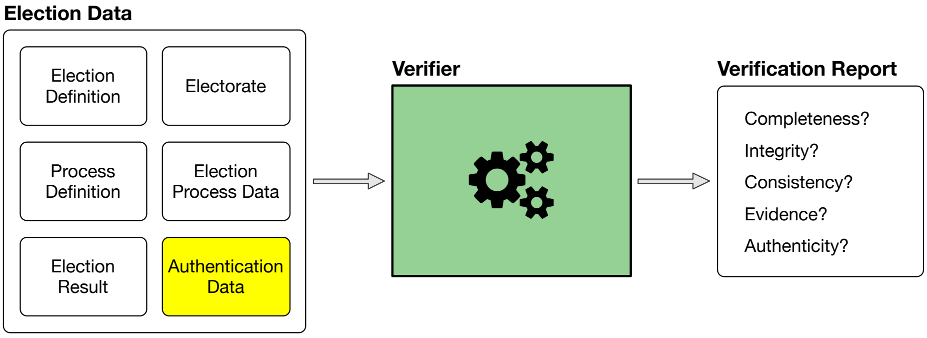


Fig. 2: Abstract model of the verification process.

By defining the verification as a series of individual tests performed by the verifier, it is possible to introduce at least five different top-level categories, according to which the tests can be grouped in a meaningful way (further meaningful categories and sub-categories may exist in more concrete cases). In Table 1, we summarize the meaning of these categories and their differences. One of the purposes of introducing such test categories is to facilitate and systematize the definition of a suitable *test catalog*, which ultimately leads to a fully connected verification chain. This test catalog is the main content of the verifier specification (see Section 2.4). Another purpose of introducing categories is to simplify the organization and presentation of the test results in the verification report.

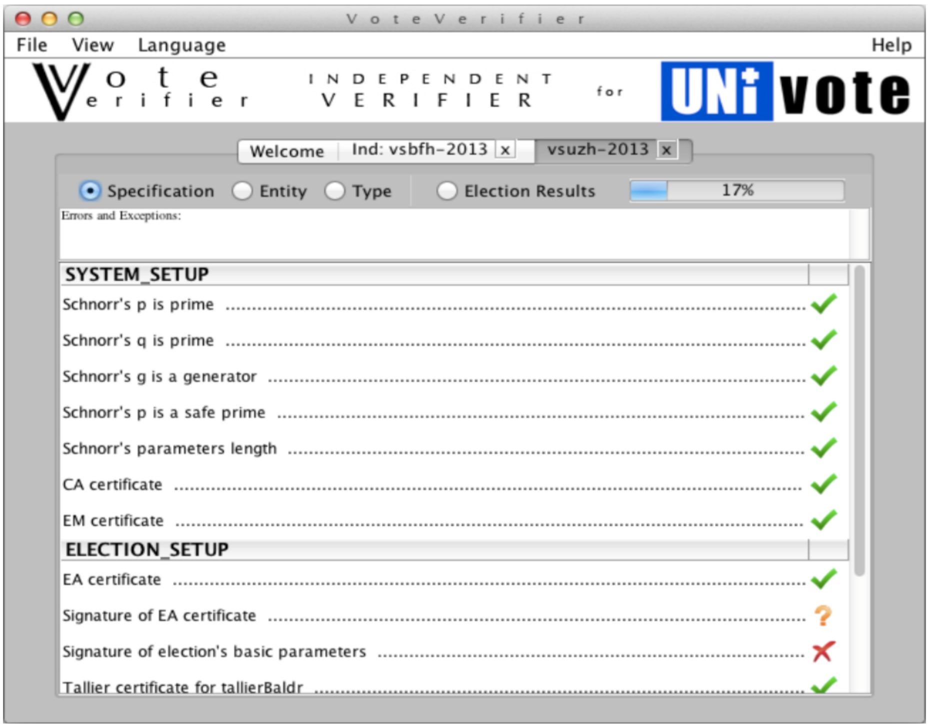
Category	Description
Completeness	Do the available data elements cover the whole election process according to the specification? Do they allow a complete verification chain?
Integrity	Do all data elements correspond to the protocol specification? Are they all within the specified ranges?
Consistency	Are related data elements consistent to each other?
Evidence	Are the cryptographic proofs contained in the election data all valid? Do they provide the necessary evidence to infer the correctness of corresponding protocol steps?
Authenticity	Can the data elements be linked unambiguously to the party authorized to create them?

Table 1: Test categories of the verification process.

An example of a verifier's user interface is depicted Figure 3. It shows the upper part of the verification report for an election at the University of Zürich in 2013 using the UniVote system [4, 10]. The status bar in the upper right corner indicates that the verification is still in progress. The report also shows the results of the first eleven (out of 61) tests. Nine tests succeeded, one test has been dropped due to a missing certificate, and one test failed due to an invalid signature. Assuming that the verifier itself works properly, this indicates that parts of the implemented voting system have not been working properly, or even worse that the election has been exposed to an attack. In any case, it is clear that both the cause and the impact of the exposed problems have to be investigated. Triggering such an investigation in case of irregularities is the main purpose of performing the verification.

2.3 Impact and Consequences of Failed Tests

As illustrated by the above example, using a verifier to conduct the verification of an election can always lead to a situation, in which some tests from the test



● Specification ○ Entity ○ Type ○ Election Results 17%

Errors and Exceptions:

SYSTEM_SETUP

Schnorr's p is prime ✓

Schnorr's q is prime ✓

Schnorr's g is a generator ✓

Schnorr's p is a safe prime ✓

Schnorr's parameters length ✓

CA certificate ✓

EM certificate ✓

ELECTION_SETUP

EA certificate ✓

Signature of EA certificate ?

Signature of election's basic parameters ✗

Tallier certificate for tallierBaldr ✓

Fig. 3: User interface of the verifier for the UniVote system.

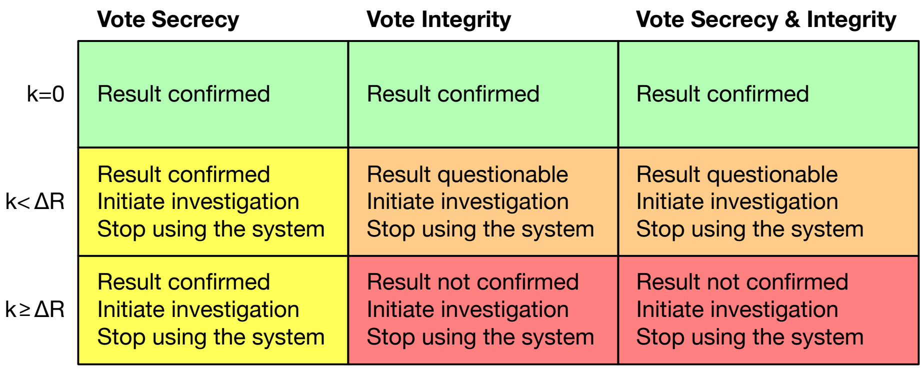
catalog have failed. There are numerous possible causes for a test to fail, but there is presumably no better way for finding the cause than analyzing the particular problem at hand. Giving general recommendations about handling failure cases is therefore quite difficult. Nevertheless, we can at least propose three different evaluation criteria, which may help to classify the impact of the problem and to decide about the next steps.

The first criteria is the maximal number of affected votes. Suppose that N electronic votes have been submitted and that maximally $0 \leq k \leq N$ votes are affected by the problem.[1] Note that this constraint includes the two natural limiting cases of $k = 0$ (no vote affected) and $k = N$ (all votes affected). Another important quantity is the number ΔR of votes, which are necessary to change the winner or the outcome of an election. In a referendum, the general constraint for this number is $1 \leq \Delta R \leq \frac{N}{2}$. For example, for 60 *yes*-votes, 30 *no*-votes, 10 blank votes, and therefore $N = 100$, the outcome could be changed by turning 15 *yes*-votes into *no*-votes. For judging the impact of the problem, it is therefore important to determine if k is smaller or bigger than $\Delta R = 15$. For $1 \leq k < \Delta R$, the impact of the problem may not justify the invalidation of the whole election (similar arguments are used to handle minor irregularities in

[1] In a hybrid election process, both the number of electronic votes and the total number of votes must be taken into consideration.

paper-based elections), but in the more severe case of $k \geq \Delta R$, repeating the whole election can probably not be avoided.

The second criteria refers to the security goal violated by the detected problem. Relative to a single submitted vote, three cases must be distinguished: a violation of the vote's secrecy, a violation of the vote's integrity, or a violation of both the vote's secrecy and integrity.[2] Generally, we consider violations of the vote's integrity to be more critical than violations of the vote's secrecy, because they affect the election results in a direct way. The possible consequences are therefore more drastic in such cases. In Figure 4 we give an overview of the consequences in the scenarios obtained from combining the first two evaluation criteria. It shows for example that vote secrecy violations do not directly invalidate the election result, but that an investigation of the problem's cause is always necessary.



	Vote Secrecy	Vote Integrity	Vote Secrecy & Integrity
k=0	Result confirmed	Result confirmed	Result confirmed
k< ΔR	Result confirmed Initiate investigation Stop using the system	Result questionable Initiate investigation Stop using the system	Result questionable Initiate investigation Stop using the system
k≥ ΔR	Result confirmed Initiate investigation Stop using the system	Result not confirmed Initiate investigation Stop using the system	Result not confirmed Initiate investigation Stop using the system

Fig. 4: Problem scenarios with consequences.

Another important point to consider in case of an unsuccessful verification is the question of whether the problem could possibly be solved by repeating some steps of the election process. For example, the case of a missing or invalid signature could possibly be solved by simply repeating the signature generation. Generally, such recovery procedures mostly exist for data that is not temporarily linked to other parts of the election data. In those cases, only the availability of the data is necessary to conduct the verification, not their moment of creation. Problems encountered with such data can therefore be solved by repeating their creation during a recovery procedure.

Assuming that recovery procedures exist, pursuing them will always be the first choice in case of encountering a problem in the verification report. A general business process model for handling failure cases is depicted in Figure 5. It shows that executing a recovery procedure invokes an additional verification

[2] The main purpose of the universal verification is detecting integrity violations. However, the failing of certain tests can also lead to situations, in which vote secrecy is no longer guaranteed, for example if the signatures of the mixing proofs are all invalid. This could mean that all mixing proofs have been generated by the same party, which can then establish links from cleartext votes to voters.

round. If the problem persists—or if no recovery procedure has existed from the beginning—then an investigation of the problem must be invoked and the result of the investigation must be documented in a report.

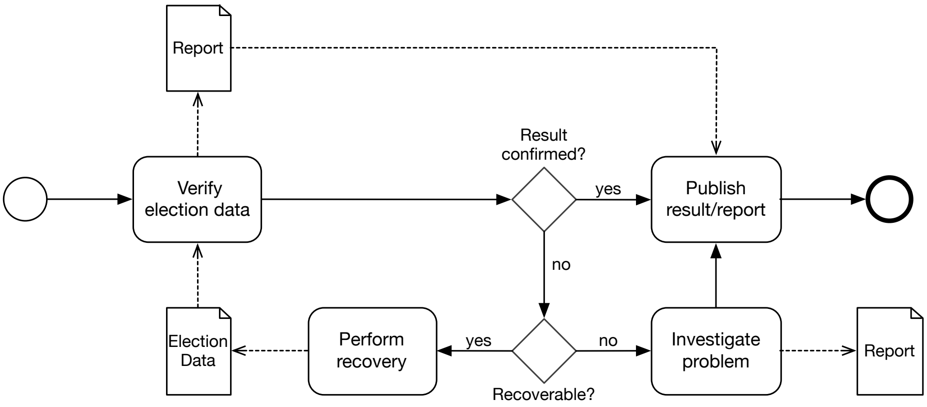


Fig. 5: Process model for handling failure cases and recovering from them.

2.4 Developing the Verifier

The principal technical aid for conducting the verification of an election is the verifier. Clearly, the proper functioning of the verifier is a mandatory precondition for obtaining conclusive verification reports. It is therefore essential that the verifier works exactly in accordance with the specified cryptographic protocol. Any deviation could lead to unpleasant situations in which the verifier reports a failure when everything is correct (false negative) or misses a failure when something went wrong (false positive). In a nutshell, the software development goal for the verifier consists in avoiding these situations altogether.

Given the mathematical and technical complexities of cryptographic voting protocols, developing a verifier directly from the protocol specification is a very big challenge. It requires advanced skills in both applied cryptography and software development. If unqualified personnel is in charge of this task, it is likely that the implemented test catalog will not form a complete verification chain, or that some tests are implemented incorrectly. In both cases, the conclusiveness of the verification report is weakened considerably.

To ensure the required functionality and software quality, we propose a two-step procedure for developing the verifier. The first step consists in deriving a specification document from the specification of the voting system. This task should be performed by cryptography experts that are familiar with voting protocols in general and with the specific technical details of the voting protocol at hand (possibly by the designers of the voting protocol). The main part of this document is the aforementioned test catalog, which together must form a complete verification chain. To assure the completeness of this chain, assembling the test catalog must be carried out with meticulous precision. To detect remaining gaps as early as possible, we also recommend applying a thorough reviewing process to this document. For maximal transparency, we also recommend the publication of this document.

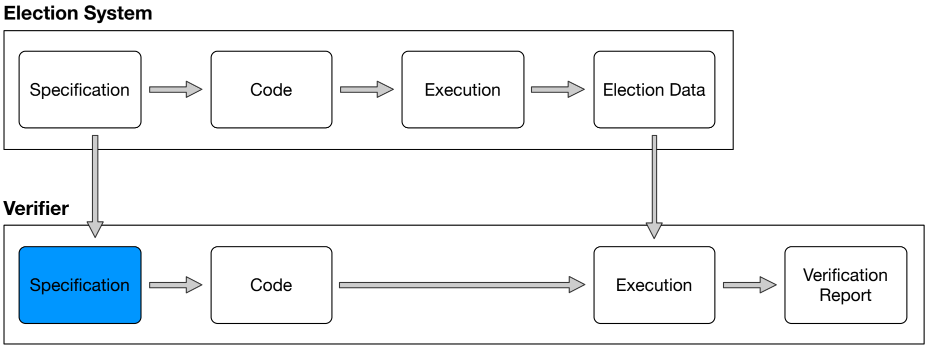


Fig. 6: Developing and executing the verifier based on a separate specification document.

Developing the actual software based on the verifier's specification is the second step of the proposed procedure (see Figure 6). This task can be delegated to a software engineer with only moderate background knowledge in cryptography and cryptographic protocols. To achieve general software quality properties, standard software design and coding principles should be applied to the development process. Code reviewing is another important method to establish the desired code quality. For maximal transparency, we also recommend to publish the source code and to invite the public to participate in reviewing the code.

Additional preconditions for developing the verifier are a precise interface description for obtaining the election data from the voting system and the availability of some meaningful test data. Both preconditions must be met by the developers of the voting system. Ideally, the test data also contains inconsistencies or flaws, such that the developed software can be tested for false positives and false negatives. Finally, it is also very important to implement a strict versioning policy, because even the slightest change in the voting system or in the election data may be enough to affect the proper functioning of the verifier.

3 Hybrid Election Processes

The election and verification processes as discussed so far are only directly applicable to the simple case of a purely electronic election with a single voting channel. The situation usually gets more complicated if multiple voting channels are offered simultaneously. The simplest way of handling multiple channels is to let the voters choose their preferred channel prior to an election. This leads to a decomposition of the electorate, which finally results in conducting multiple elections independently of each other. In this case, no channel coordination other than summing up the individual election results is necessary. If one of the channels is an electronic one, the verification can therefore be conducted in isolation using the process described in the previous section.

A more complicated situation arises if voters can choose the voting channel spontaneously during the election period. The composition of corresponding election processes is called a *hybrid election process*, and we will see in this section that extra precautions are necessary to handle this case properly. We are particularly interested in hybrid election processes because they correspond to the current plans in Switzerland of offering the electronic channel in addition to the two existing voting channels (postal mail, in person). The question that we want to address here is how to conduct the verification of the electronic votes, if postal voting or voting in person takes place simultaneously.

3.1 Extending the Election and Verification Processes

The major problem that arises in a hybrid election process is to ensure that no voter submits more than one vote over the available channels. This implies that using one channel for submitting a vote must disqualify the voter in every other channel. It is clear that implementing this seemingly simple principle requires accurate coordination between the channels. In practice, it turns out that the submission of multiple votes over multiple channels can not be avoided completely, even if doing so is illegal. If this happens, it should at least not be possible that two votes from the same voter are counted. Double votes from the same voter must therefore be eliminated—together with other invalid ballots—before starting the tallying process. This process, which is called *cleansing*, is a mandatory initial step of the post-election phase.

From the perspective of the election process model of Section 2.1, an additional input containing the list of disqualified voters is required to perform the cleansing of the submitted ballots before initiating the tally. This leads to the extended election process model of Figure 7. The actual electorate that is relevant for the tally is obtained from eliminating the disqualified voters from the electorate. The model depicted in Figure 7 also shows that the list of actual election participants is an additional output of the process. This list defines the disqualified voters in every other voting channel of the hybrid system. In the next subsection, we will see how to combine two or multiple such election processes into a hybrid election process.

The additional input and output documents in the extended election process model must be taken into account when performing the verification. Note that every single entry in each of these documents is highly critical, because they define somebody's right to submit a vote over some channel. Figure 8, which shows the extended verification process model, illustrates the inclusion of these documents. The purpose of the verification report is still the same, but since two additional inputs are now taken into account, a successful report also validates their contents.

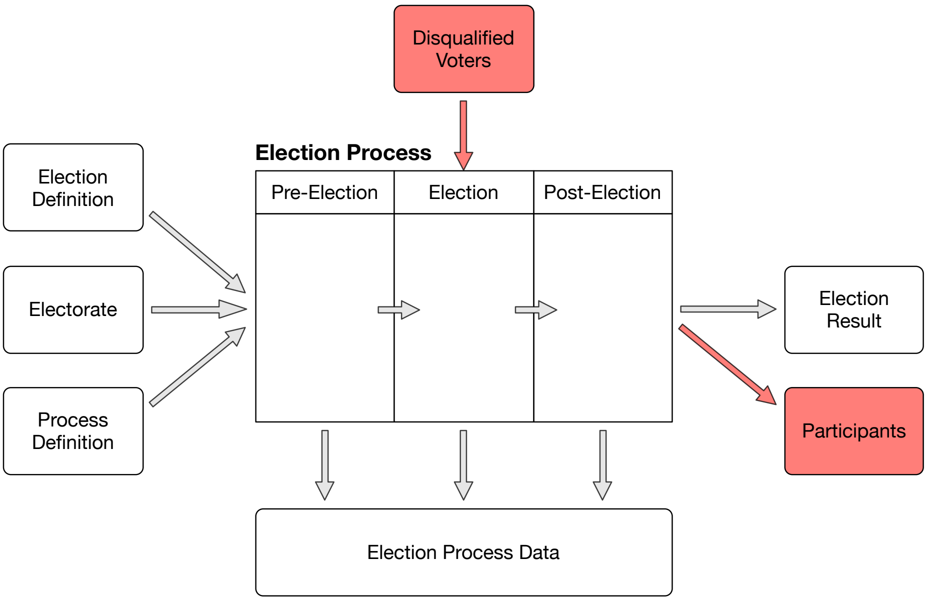


Fig. 7: Extended election process model for hybrid elections.

3.2 Composed Election Processes

Let's now have a closer look at actual compositions of multiple election processes. We will restrict ourselves to the simplest case of composing two alternative election processes. As we will see, the result of such a composition is again an election process, which can be further combined with other election processes. In this way, it is possible to construct recursive process models for more complicated combinations of three or more voting channels on the basis of the basic compositions described here.

For analyzing the composition of two election processes, we can distinguish two opposed cases. In the case of a *serial composition*, the temporal availability of the two channels is exclusive, i.e., the election period of the first election process strictly precedes the election period of the second process. Figure 9 depicts the hybrid process model obtained from a serial composition. It shows that the list of participants from the first channel defines the list of disqualified voters in the second channel. Note that the inverse data flow from the second channel back into the first channel is not required to guarantee the detection of double votes. Serial compositions are therefore relatively easy to handle properly.

More complicated situations arise in the case of a *parallel composition*, in which the election periods of the two processes overlap. In this case, the data exchange between the two channels is mutual. The resulting process model is depicted in Figure 10. It shows how the list of participants from each of the two channels is given as an additional input into the other channel. The problem here is that the same voter may appear in both lists, which must be taken into

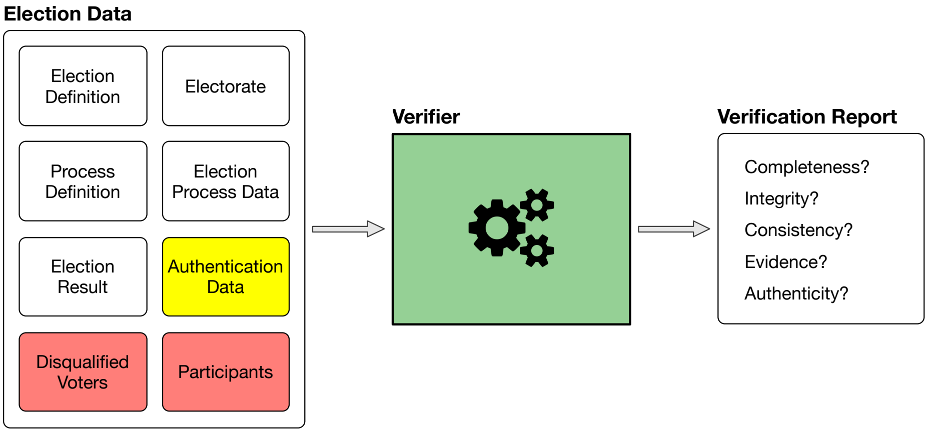


Fig. 8: Extended verification process model for hybrid elections.

account in each of the two cleansing processes. To handle such cases properly, there must be a clear policy of prioritizing one of the two submitted votes.

We see three different general strategies for defining such a policy. We will shortly discuss them in the remaining of this section. For this, we consider the use case from Switzerland, where an electronic voting channel is combined with a physical voting channel (postal mail). We assume that an electronic vote counts as "submitted" when the voter terminates the voting process, for example by clicking a button from the voting application's user interface. In case of submitting a paper ballot using postal mail, we assume that the vote counts as "submitted" when the ballot is registered at the polling station. Note that in Switzerland, submitting more than one vote is prohibited by law, regardless of the available voting channels. However, since voters are instructed to submit a paper vote in case of a problem encountered when submitting an electronic vote, enforcing this law will be difficult in practice. In other countries, for example in Norway, submitting multiple votes is explicitly allowed. In such a case, the last submitted vote overrides all previously submitted votes.

Prioritizing the Physical Channel. The rule here is as follows: if the same voter uses both channels to submit a vote, only the vote submitted over the physical channel will be counted. With this policy, paper votes can be counted regardless of the list of participants from the electronic channel. Therefore, the problem of eliminating double votes is only relevant for the electronic channel. Note that this situation is similar to a serial composition, in which the physical channel precedes the electronic channel. This policy is therefore relatively simple to implement. It is also compatible with a current practice in Switzerland, where administrative staff at the electoral office separates paper votes from the signed polling cards right upon receiving the paper ballot.

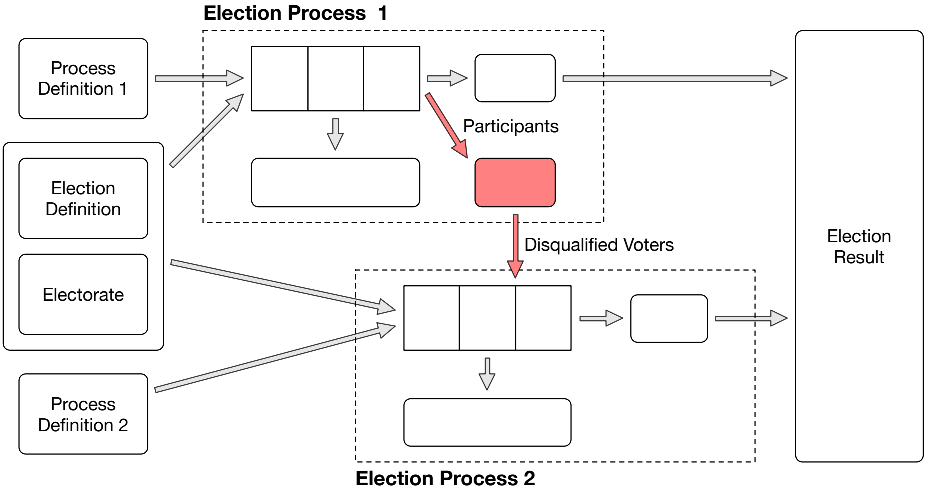


Fig. 9: Serial composition of two election processes.

Prioritizing the Electronic Channel. Here, the rule from above is applied in the opposite way, i.e., only the electronic vote of a voter using both channels is counted. The counting of the electronic votes can therefore be conducted regardless of the list of participants from the physical channel. This also simplifies the verification process, which can be conducted independently of the physical channel, but it makes the counting of the paper ballots at the polling station more complicated. For example, separating the paper votes from the signed polling cards must be postponed until the complete list of participants from the electronic channel is available.

Prioritizing the First or Last Submitted Vote. In this case, if someone submits two votes over both channels, only the first or the last submitted vote will be counted. This is the most complicated policy to implement, because the channels are mutually dependent on each other, i.e., exchanging both lists of participants according the Figure 10 is a mandatory precondition for eliminating double votes in both channels. The exchange of these lists can be done in two ways, either dynamically during the election phase or in a single step at the end of the election phase. In the dynamic case, the two voting channels may try to sort out double votes at the moment of receiving them, but a perfect synchronization is obviously very difficult to implement. Therefore, conducting the cleansing process at the end of the election phase is necessary in either case.

To enable the prioritization of either the first or the last submitted vote, timestamps must be added to the lists of participants, which define the exact moment of submitting the vote. The decision of keeping or ignoring a submitted vote is then based on these timestamps. Note the issuing reliable timestamps in an electronic context is a difficult problem on its own, especially if third parties must be able to verify the correctness of the timestamps in a conclusive way.

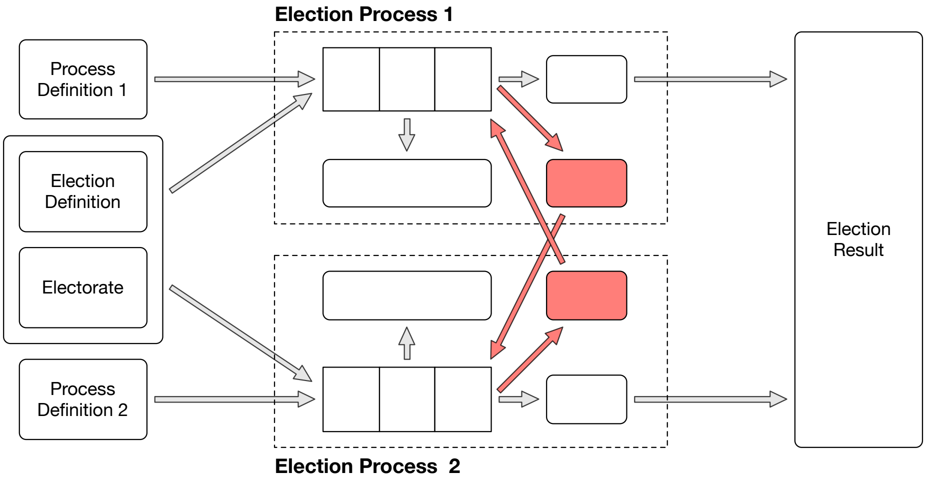


Fig. 10: Parallel composition of two election processes.

4 Conclusion

This paper is an attempt to define the universal verification process for electronic elections. The motivation for this paper comes from the observation that there is almost no practical experience with conducting actual verifications. On the other hand, since universal verifiability is commonly recognized as one of the most important counter-measures against all sorts of failures or attacks, almost everyone agrees that it must be implemented into future e-voting systems that are used for real political elections. Our analysis of the verification process in this paper shows that conducting an actual verification is more complex than it may appear at first sight. By discussing some of the most apparent questions and problems, we hope to provide some general guidelines for people in charge of implementing or organizing a verification process.

In most parts of the paper, for making our analysis and findings as widely applicable as possible, we have adopted a very general perspective. However, relative to a concrete voting system and application use case, many specific questions only arise if all the details about the cryptographic voting protocol, the technical system specification, and the political and legal contexts are available. Therefore, we can not answer these questions here, but we recommend not to underestimate the problems that may arise. More generally, we recommend to pay attention to the difficulties of the verification process well in advance. Election organizers should look at it as a separate important project, which also requires a careful planning, proper management, and adequate budget.

Acknowledgments. We thank the anonymous reviewers for their comments and suggestions. We are also grateful to Timo Bürk, Xavier Monnat, Jörg Schorr, Olivier Esseiva, and Anina Weber for helpful discussions, proofreading, and feedback. This research has been supported by the Swiss Post AG.

References

1. *Verordnung der Bundeskanzlei über die elektronische Stimmabgabe (VEleS).* Die Schweizerische Bundeskanzlei (BK), 2013.
2. B. Adida. Helios: Web-based open-audit voting. In P. Van Oorschot, editor, *SS'08, 17th USENIX Security Symposium*, pages 335–348, San Jose, USA, 2008.
3. D. Chaum, A. Essex, R. Carback, J. Clark, S. Popoveniuc, A. Sherman, and P. Vora. Scantegrity: End-to-end voter-verifiable optical-scan voting. *IEEE Security & Privacy*, 6(3):40–46, 2008.
4. E. Dubuis, S. Fischli, R. Haenni, S. Hauser, R. E. Koenig, P. Locher, J. Ritter, and P. von Bergen. Verifizierbare Internet-Wahlen an Schweizer Hochschulen mit UniVote. In M. Horbach, editor, *INFORMATIK 2013, 43. Jahrestagung der Gesellschaft für Informatik*, LNI P-220, pages 767–788, Koblenz, Germany, 2013.
5. D. Galindo, S. Guasch, and J. Puiggalí. 2015 Neuchâtel's cast-as-intended verification mechanism. In R. Haenni, R. E. Koenig, and D. Wikström, editors, *VoteID'15, 5th International Conference on E-Voting and Identity*, LNCS 9269, pages 3–18, Bern, Switzerland, 2015.
6. I. S. Gebhardt Stenerud and C. Bull. When reality comes knocking – Norwegian experiences with verifiable electronic voting. In M. J. Kripp, M. Volkamer, and R. Grimm, editors, *EVOTE'12, 5th International Workshop on Electronic Voting*, number P-205 in Lecture Notes in Informatics, pages 21–33, Bregenz, Austria, 2012.
7. R. Haenni and R. E. Koenig. Universelle Verifizierung von Wahlen und Abstimmungen über das Internet. *SocietyByte*, June 2017.
8. R. Haenni, R. E. Koenig, P. Locher, and E. Dubuis. CHVote system specification. *IACR Cryptology ePrint Archive*, 2017/325, 2017.
9. K. Häni and Y. Denzer. Visualizing Geneva's next generation e-voting system. Bachelor thesis, Bern University of Applied Sciences, Biel, Switzerland, 2018.
10. G. Scalzi and J. Springer. VoteVerifier: Independent vote verifier for UniVote elections. Bachelor thesis, Bern University of Applied Sciences, Biel, Switzerland, 2013.
11. M. Volkamer, O. Spycher, and E. Dubuis. Measures to establish trust in Internet voting. In *ICEGOV'11, 5th International Conference on Theory and Practice of Electronic Governance*, Tallinn, Estonia, 2011.

Usability is not Enough: Lessons Learned from 'Human Factors in Security' Research for Verifiability

Oksana Kulyk[0000−0003−4218−1658], Melanie Volkamer[0000−0003−2674−4043]

Karlsruhe Institute of Technology, Karlsruhe, Germany
name.surname@kit.edu

Abstract. A well-known issue in electronic voting is the risk of manipulation of the cast vote. For countering this risk, a number of methods have been proposed that enable the voter to verify that their cast vote actually represents their intention, the so-called *cast-as-intended verification*. Yet, the empirical studies on the voter's behaviour towards using these methods show that often only a small amount of voters attempts the verification or succeeds in performing it. Poor usability of the verification procedure has been often named as the main reason for such a failure of the voters to verify. Research into human factors in other security domains, however, reveals other reasons aside from poor usability, that hinder the proper adoption of security practices among end users. In this paper we discuss these factors with respect to their applicability to cast-as-intended verification. Our results indicate, that many of these factors are potentially relevant in the electronic voting context, too. Correspondingly, we conclude that additional measures aside from ensuring the usability of the cast as intended verification mechanisms are required in order to make sure that the voters successfully verify the integrity of their votes. As such, corresponding mechanisms are proposed.

1 Introduction

Remote e-voting over the Internet can solve many problems. Voters from abroad are included more easily as well as voters with disabilities. Furthermore, voting from wherever Internet is available gains attraction since a polling station does not have to be visited during particular hours and day(s). Although there are many benefits, remote Internet voting channels introduce new possibilities for adversaries that aim to maliciously influence the outcome of the election directly by changing votes or indirectly by breaking vote privacy. Therefore, Internet voting systems introduce new challenges. One of these challenges is the so-called trusted platform problem: Since the voting device typically is a voter's device, e.g. computer, laptop, tablet, or smartphone, this device is beyond the control capabilities of the election authorities and of the provider of the Internet voting system. Hence, an adversary might take control over voters' devices to maliciously manipulate the outcome of the election. Another challenge is to detect a malicious vote casting software. In this case an adversary would manipulate the vote casting software in a way that votes would be changed before storing them in the electronic ballot box.

Previous research on electronic voting resulted in numerous proposals for addressing these challenges. While some of the proposals focus on ensuring the security of the voting devices via trusted platform module [26], most of the state-of-the-art research is dedicated on proposing techniques that enable voters to verify that their vote has been sent to the voting system without being manipulated by the voting device or the vote casting software (i.e. providing *cast-as-intended verifiability*). These proposals include cryptographic protocols, as well as ready-to-use implementations of corresponding cryptographic protocols within deployed voting systems. However, even if the system provides the possibility to verify that the voters' choices have been encoded correctly, it is not guaranteed that voters actually make use of this functionality. In particular, the available statistics of elections using Internet voting systems demonstrate that a very small percentage of all voters actually verifies [4, 10]. One of the reasons for such a low number is the fact that verifying is not usable enough, too complicated, and confusing for the voters.

Outside of electronic voting, the research of human factors in security mechanisms has identified and studied various factors besides the usability of security mechanisms that prevent users from protecting their security and privacy by applying corresponding mechanisms. Whether these factors are applicable for verifying votes has not been considered in electronic voting research, yet. Thus, the goal of this paper is to analyze whether selected human factors identified for security mechanisms in general, are applicable for the security mechanism 'cast as intended' verification. We focus on the following directions:

General factors: We discuss the relevance of corresponding factors such as lack of awareness risks identified for security and privacy mechanism in [27]. We decided to go for this paper as the factors are identified based on an interview study and a literature review. We discuss the applicability of their factors for cast as intended verification.
Psychological factors related to social engineering attacks: We discuss the factors identified for success of social engineering attacks in other cyber security contexts, i.e. the adversary relying on the victim's tendencies to obey the authority. In our discussion we rely on [31]. The authors derived factors via an empirical study.
Attacks focusing on the user interfaces: We discuss how an adversary can modify interfaces in a way that the security mechanism disappears or gets very un-usable.

We show, that most of these factors are applicable for 'cast as intended verifiability'. As such, while the usability of the proposed solutions plays an important role, other factors such as the lack of awareness of security threats need to be addressed. Furthermore, we discuss the implications from these findings for the future of electronic voting.

2 Background and Related Work

In this section, we describe previous work on cast-as-intended verification methods and the research on human factors in the verification.

2.1 Methods for Cast-as-Intended Verification

A number of methods for cast-as-intended verification have been proposed in the literature. The most prominent examples of methods used in voting systems are as follows (see also [8] for a more detailed taxonomy):

Decryption-Based: In order to verify that her vote was encrypted and cast correctly, the voter uses a second device (the so-called verifier) such as a smartphone. The randomness used for encrypting the vote is transferred from the voting device to the verifier. The verifier uses the randomness to encrypt each one of the available voting options and compare the resulting ciphertexts with the encrypted vote sent to the voting server. As soon as the match is identified for one of the voting options, the verifier outputs the corresponding voting option to the voter, who in turn verifies that the option matches her intent. This approach, in particular, is used in the Estonian system [9].
Challenge-or-Cast: A variant of the decryption-based method, the challenge-or-cast verification also requires using an external verifier, which is either a second device [19], a website of the trusted institution [19] or software running on the voter's device [2]. The main difference to the decryption-based approach is that after the vote is encrypted and the encrypted vote is output to the voter, the voter chooses either to cast it or to challenge the voting system. In case the voter chooses to challenge, the randomness and the chosen voting option are transferred to the verifier. The verifier encrypts the voting option using the randomness and outputs the resulting ciphertext to the voter, who finally has to compare the ciphertext with the encrypted vote output by the voting client. Once challenged, the vote cannot be cast, and the voter has to start the vote casting process again. The challenge-or-cast approach is used in the Helios system [2].
Return Codes: In this approach the verification relies on code sheets, distributed to the voter via an out of band channel (e.g. traditional mail), see e.g. [5]. The code sheets contain a list of voting options with a unique code assigned to each option. After casting the vote, the voting system outputs a so-called return code, which the voter has to compare with the code on their code sheet for their chosen option. This approach, in particular, is used in the Neuchatel voting system [7].

2.2 Human Factors in Cast-as-Intended Verifiability

A number of works explore the human factors involved in the cast-as-intended verification. These works, in particular, focus on the following research questions: whether the verification process itself is effective (i.e. whether the voters are capable of performing the verification if they choose to do so), and whether the mental models of the voters are accurate (i.e. whether the voters understand the concept of verification well enough to be motivated to verify). The usability in terms of effectiveness of the cast-as-intended verification has been the focus of various studies. As such, the study in [6] evaluated the usability of the Norwegian Internet Voting system, identifying usability shortcomings in the verification process. Similarly, the usability of cast-as-intended verification in the Helios voting system has been evaluated in several studies [1, 11, 15, 29], revealing that many of the study participants were not able to perform the verification successfully. Various modifications of the verification process in Helios have furthermore been investigated via a user study [15], revealing that although these modifications managed to improve the usability of the original proposal, further problems remain that prevent the participants from successfully verifying. The studies conducted by Acemyan et al. [1] furthermore evaluated the usability of the Pret a Voter and Scantegrity II voting systems, concluding that the usability of the verification in these systems was poor as well. The usability of various approaches for cast-as-intended verifiability has been investigated by Marky et al. [16] via an expert evaluation approach based on cognitive walkthrough method. The investigation revealed a number of assumptions on voter capabilities, such as the ability of the voters to compare random-looking strings of characters, cruical for ensuring the security of the investigated approaches. Other studies focused on the mental models of voters regarding verifiability in electronic voting. As such, the study of Olembo et al. [22] identified five groups of mental models (Trusting, No Knowledge, Observer, Personal Involvement and Matching), revealing that the voter's understanding of verifiability is often lacking and thus preventing the voters from performing the verification. The follow-up study [21] furthermore evaluated the effect of diverse messages in motivating the voters to verify, revealing further misconceptions regarding the verification process, prevalent among the voters and preventing them from verifying, such as beliefs that their experience as a computer user is enough to protect against possible vote manipulation. Further misconceptions prevalent among the voters regarding the verification were revealed by the study of Schneider et al. [23], i.e. the belief that the verification is only needed to safeguard against voter's own mistakes (such as accidentally choosing the wrong candidate) as opposed to malicious vote manipulation.

3 General Factors

The factors preventing end users from adopting secure behaviour and from using available solutions for security and privacy protection have been investigated in various contexts, such as smartphones [27] or password managers [3]. These works have shown, that while many of the investigated solutions lack in usability, there are other factors no less important for end user to adopt these solutions. As a systematization of these factors, a model has been proposed by Volkamer et al. [27], distinguishing between the different factors that the developers of security mechanisms need to address. These factors are: *lack of awareness*, *lack of* *concern*, *lack of self-efficacy*, *lack of compulsion* and *lack of perseverance*. In this section, we elaborate on each factor and its possible implications in the electronic voting context for cast-as-intended verification.

3.1 Lack of Awareness

According to [27], many users don't see a need to use security mechanisms simply because they are unaware of potential risks in general and specific attacks related to the corresponding mechanism.

This factor is likely to influence the likelihood that voters use the cast as intended verification mechanisms: Voters might be simply unaware of possible risks of vote manipulation that the cast as intended verification mechanism can protect against. As far as we are aware of, neither mass media nor election organizers communicate such risks. While a lot of recent media attention has been dedicated to the potential manipulations of election results with means of cyber warfare (see e.g. [33]), the discussion focused on the manipulation of components of the voting system in controlled environment, e.g. the voting machines at the polling place. The dangers to the manipulation of the vote casting software on voters' individual devices, however, has not been the focus of attention. On the opposite, several studies on voters' perception of verifiability in electronic voting [22, 23] have shown that participants' first thought is that the election management boards are responsible to select a 'secure' system and prevent manipulations. Thus, unless the voters understand the inherent necessity of verifiability for the security of voting systems, it is not very likely that they actually verify their vote.

3.2 Lack of Concern

The next identified factor is the perception regarding security and privacy risks, that while people are aware in general that these risks exist, they do not present a great concern for them personally. For instance, they are aware of phishing attacks in general but are not concerned that a phisher may attack them personally. As such, the users tend to believe, that (1) they are not important enough to become a target of the adversary, or that (2) e.g. they have nothing to hide, therefore, they should not be concerned if someone hacks into their phone. The lack of concern of end users is often misguided due to underestimating the value of personal data and overestimating the effort from hackers or service providers required to collect it or install malware, and it can – at least partially – be rationally explained: Indeed, it is not unreasonable to assume that the private communication of regular citizens is of less interest to hackers, than the private communication of high-profile politicians .

So in case of the trusted device problem, one should be careful in explaining this problem to voters. In case it is purely that voters' devices might have malware installed (not necessary for the election, but in general), the 'lack of concern' factor might be applicable for electronic voting, too. Voters might consider them as not important enough that someone installs malware on their device in general. Some of the voters therefore might conclude, that the probability of them becoming a victim of the hacker attack is low. As a consequence they are not very likely to apply cast as intended verification mechanisms.

If voters are made aware that it is important to verify to make sure their vote cannot be manipulated (using voting specific attacks) without the manipulation being detected, the applicability of the 'lack of concern' factor depends on voters' understanding of demographic elections. In democratic societies, the value of each vote counts equally[1], therefore, any citizen is equally likely to be targeted for vote manipulation, regardless of their social status. It is therefore reasonable to assume that the importance of one's vote is self-evident to many of those who choose to participate in the election (otherwise they would abstain)[2].

3.3 Lack of Self-Efficacy

The next identified factor that prevents users from adopting these solutions is the non-accessibility of the security mechanisms. As such, while the users might be aware about the risks to their security and privacy and even be concerned about the corresponding threats, they don't apply corresponding security mechanisms as they have only an abstract idea about the security mechanisms and as such (1) either consider them as being to complex for them to be used or (2) as being too ineffective (2a) against really powerful players like Google or national security agencies, or (2b) as they still need to relay on third parties taking care of their security duties (thus feeling helpless). Thus, users without technical knowledge do not have the confidence that they can apply these countermeasures effectively and/ore that the measures they can take only slightly increase the security. As a consequence they don't use the security mechanisms.

The 'lack of self-efficacy' is also applicable for the voting context and in particular for the cast as intended verification mechanisms: Voters' might not properly be aware of the mechanisms as such, they might consider it as too complicated and being afraid that they cannot properly apply it. The complexity of the verification process can furthermore discourage the voters from verifying. As shown in Section 2, it is well-known that many of the existing voting systems fail to provide such simplicity, hence, the voters might feel overwhelmed even before they attempt the verification.

Furthermore, they may consider the mechanisms as too ineffective as taking the steps is useless if others are not taken by the voting system company, the election management boards, and the crypto experts who for instance take care of other verification issues including eligibility verification as well as the system's availability. Furthermore, voters may consider the mechanisms as useless as the

[1] While there might be inequalities between the weight votes from different districts in some political systems, e.g. via so-called gerrymandering, the equality still holds among the voters within a district.

[2] Note, however, that the issue might be less clear within the countries that have mandatory voting, as some voters might vote in order to avoid penalty rather than believing that their vote has an effect on society.

cast as intended mechanism might not offer protection against a very powerful adversary (e.g. the one who can break the cryptography behind the method, or corrupt both the voter and the verification device). They might not see that the mechanisms still provide a level of security sufficient for many cases.

3.4 Lack of Compulsion

Lack of compulsion has been identified as another factor in preventing the adoption of security mechanisms: Even if the users recognise that there is some value in these mechanisms, this perceived value is still outweighed by the costs of adopting the mechanisms, such as time, effort, but also possible financial costs. For example, inconvenience caused by these mechanisms, e.g. by having to input the password in order to unlock the smartphone each time one wants to use it, or the performance drop caused by installing an antivirus, have been commonly named by the users who decided against using these protection measures.

With respect to the applicability of this factor for cast-as-intended verification mechanisms, it is important to mention that elections don't happen too often, even in countries with relatively frequent elections such as Switzerland. Even if there are elections every three months this is not as often as unlocking a smartphone. So time might on the one hand not play such an important role. On the other hand most people when thinking of casting their vote online, think of a simple solution such as logging in, selecting a candidate, confirming the candidate and that's it, compared to shopping online. These issues were shown by previous studies to be prevalent among existing voting systems, as described in Section 2. Such mental models about vote casting processes make the factor 'lack of compulsion' applicable for cast as intended verification mechanisms in particular in cases the steps should be repeated several times as required with the Benaloh Challenge. However, previous studies show that the voters would be ready to use systems that require more time and effort from the voter for both vote casting and verifying, if these systems provide a higher level of security and this higher level of security is made transparent to them [13]. What has not been studied – to the best of our knowledge – whether voters would be willing to take any costs for verifying (e.g. a special device).

3.5 Lack of Perseverance

The last identified factor addresses the lack of perseverance: i.e., even among the users who are generally willing to adopt more secure behaviour, many still get side-tracked and therefore fail to make such behaviour as long-term habit in their daily life. One of the named reasons in the security context is the fear of social pressure and of appearing paranoid by paying too much attention to security.

As the verification procedure is meant to be performed by each voter on her own, one can presume that the social pressure aspects are less likely to play a role in the voters' desire to verify their votes. Yet, if the attitude prevalent in the society is that the verification option only exists to appease the minority of most concerned voters, and the rest do not need to verify – an attitude which presence was confirmed by previous studies [23] – this could negatively affect the voters desire to verify.

4 Psychological Factors

In the field of security, a number of attacks have arised, that aim to 'manipulate' the end users or administrators via deception techniques known as social engineering. The previous research by Workman [31, 32] identifies the following psychological factors contributing to the success of such attacks: *trust*, *normative commitment*, *continuance commitment*, *affective commitment* and *obedience of authority*. In this section, we briefly explain the identified psychological factors and discuss their applicability in the context of cast as intended verification to execute corresponding social engineering attacks.

4.1 Trust

The first psychological factor is the willingness of users to trust: Many of the attacks relying on social engineering exploit the general willingness of the user to trust. As such, the adversary attempts to appear trustworthy to their victims, for example, by pretending to be someone from their social circle, so that the victim would comply with the attacker's request, such as granting the adversary access to the system.

In the context of electronic voting, trust would be gained in case the adversary spoofs the email address of either the election management board, the party the candidate is a member of or in favour of, or some other parties being officially involved in the process such as the vendors, international observers, or the security experts. Having this in mind very easy deployable social engineering attacks are of interests, i.e. sending so called phishing emails reminding people to vote but including the slightly change URL in the email. Depending on the voting system in place the phisher would be successful with this approach or not. It is most likely that this approach is only successful in combination with others (see e.g. Section 5).

The willingness of the voters to trust the adversary can be particularly exploited in the systems that rely on external verifiers. Such systems, in particular, include either explicitly delegating the verification to a third trusted party [19,24], or letting the voter to choose and install the verification software from such a party [19]. In case an adversary manages to masquerade themselves as a trusted third-party to the voters, they can subvert the verification of these voters. Thus, the voter believes to verify with the support of a trustworthy party but actually the verifier is not trustworthy. Actually, in this case voters would adopt the mechanisms but it would not mean that their vote cannot be altered.

4.2 Normative Commitment

The second identified factor is called 'normative commitment'. The social engineering attacks exploiting normative commitments rely on the person's feeling of obligation towards the attacker, for example, by offering a pay-off to the victim in exchange for a favor for the attacker, e.g. as part of a game to see whether people provide passwords for a chocolate bar.

In the context of electronic voting, an example of attacks relying on normative commitment would be vote buying. It remains, however, an open question, whether social engineering attacks exploiting the normative commitment factor in the context of verification specifically are possible. It looks like, this factor is not an issue for the adoption of cast as intended verification.

4.3 Continuance Commitment

The attacks exploiting the continuance commitment factors, according to [31], rely on the costs and benefits of an action as perceived by the victims. As such, these attacks aim to persuade the victims, that the effort required to take precautionary security measures (a) outweighs the risks that these measures are designed to protect against and (b) in particular because taking security measures comes with increased privacy risks.

In the context of electronic voting, the attacker might want to exploit the fact, that the verification procedures require additional effort from the voters, and persuade the voters into not verifying by downplaying the risks of vote manipulation. As such, the attacker might convince the voters that the voting system is trustworthy enough without the need of additional verification, or that there is no need to verify for each voter and it is enough that the security experts do verify. The question here is how to distribute this information. The success rate clearly depends on the measures taken to make sure voters understand the importance of the cast as intended verification mechanism.

Another measure the adversary can take is hyperbolising the costs of the verification to the voters. As such, the attacker can rely on lack of voters' knowledge about the security properties of the verification procedure and convince the voters that performing this procedure leads to certain security risks. An example of such an attack would be convincing the voters that as soon as they verify, the voting system will know how they voted, leading to loss or decreasing the level of vote privacy. Not having a cryptography background makes it likely to believe in this. For example, in case the return codes are used for the verification, the voter might think that it is impossible for the system to output the right return code without knowing how the voter has voted. With challenge-or-cast verification which uses an external verifier, the voter might think that the verifier knows the option chosen by the voter, without realising that only the challenged vote (which might be different from the actual cast vote) is revealed to the verifier.

4.4 Affective Commitment

Attacks exploiting affective commitment rely on the feeling of emotional ties of the victim with the group the attacker claims to represent. Such attacks, in particular, can be performed via social networks, whereby the attacker might try to pretend to be someone connected to the victim's social circle and persuade the victim into divulging private information.

In the context of electronic voting, the attacker might try to exploit the positive attitudes of the voters towards the groups that advocate using the proposed voting system, such as the state or the political parties who proclaim themselves to be in favor of electronic voting. In such a scenario, an adversary might manage to convince the voters that they do not need to verify their vote, since they do not doubt the integrity of the system. Such attacks can be particularly successful if the voters who choose to cast their vote electronically already tend to have more trust in the government than the voters who prefer more traditional means – a correlation that has been supported by some of the previous studies [18].

4.5 Obedience of Authority

Many of the social engineering attacks involve the attacker taking the role of the person of authority to the victim, with the goal to make the victim to comply to the attacker's requests.

In the context of electronic voting, attacks exploiting the voters' obedience towards authority would be based on voter coercion, e.g. as threats from an authority figure to vote in a specific way. It is, however, to be determined, to which extent the verification process can be targeted.

5 Attacks Focusing on the User Interfaces

Providing means for cast-as-intended verification, implementing them via usable interfaces and addressing the above mentioned factors, however, is not sufficient for reliable election results. Even if the original interfaces are usable, an adversary could manipulate them in a malicious way in order to prevent the voter to carry out verification successfully. These attacks can exploit two possibilities: modifying the design of the verification interface, or modifying the verification process as displayed to the voter.

Modifying the Design. Poor usability of the design of the verification can lead to voters failing to perform the verification, for example, by not knowing which button to click, as shown by previous studies [11, 15]. As such, a number of heuristics for developing usable interfaces has been developed, in the field of human-computer interaction in general [20] (e.g. providing sufficient feedback and status information to the user), as well as in security context specifically [14] (e.g. presenting the security-relevant information in an abstract way instead of confronting the user with technical descriptions). Following these heuristics, hence, the developers of voting system interfaces can potentially improve the voters' capabilities for performing the verification successfully and reduce the time and effort required from the voters to do so. On the other hand, an adversary controlling the voting software can modify interfaces in the opposite direction, deliberately making the verification non-usable, for example, by making the important elements on the web page less visible to the voters or even blocking them entirely. Such an attack would be unnoticed by the voters, unless they had the chance to familiarise themselves with the interfaces earlier.

Modifying the Procedure. In case the voters are not aware of the proper verification procedure, an adversary might use this lack of knowledge and alter the procedure. This concerns the steps after a voter chooses to verify. As such, in systems that require the voter to explicitly start the verification (e.g. Helios), the adversary could display a success message direct after the voter clicks on the "Verify-"Button. In case the voter does not know what to expect, she would assume a successful verification procedure although, she has not verified at all. A similar attack can be conducted on the systems that integrate the verification into the vote casting process, namely, the systems based on return codes. As such, an adversary can display a finalization message directly after vote casting. If the voter does not know that a return code is expected, they would not perform the verification.

6 General Discussion and Conclusion

In this section we describe the implication of the results from the previous chapters, proposing countermeasures to adress the derived factors and outlining further possible directions of future work.

6.1 Countermeasures

The presented research in human factors in security shows clearly, even if the voting system provides means for cast-as-intended verifiability and the steps are even usable, a number of factors can prevent voters from verifying. Hence, measures for addressing these factors should be taken. We describe the necessary steps in this section and discuss possible consequences of applying these steps. An overview of the proposed countermeasures and the factors they address is provided on Figure 1.

Raising Awareness for Risks and the Verification Procedures to Mitigate the Risks. As shown in Section 3.1, the voters' lack of awareness of possible vote manipulation can prevent them from verifying their vote. Furthermore, as shown in Section 4.3 an adversary can try to deliberately downplay the risks, thus convincing the voters that the verification is not necessary. Hence, measures should be taken to ensure that the voters are aware about the possibility of vote manipulation via compromised voting client software or vote casting platform.

Note, confronting the voters with the risks of manipulated votes and the fact that only by verifying, these risks can be mitigated effectively, a plausible reaction would be that this particular electronic voting system should not be used and the election management board should only use a system which is secure enough

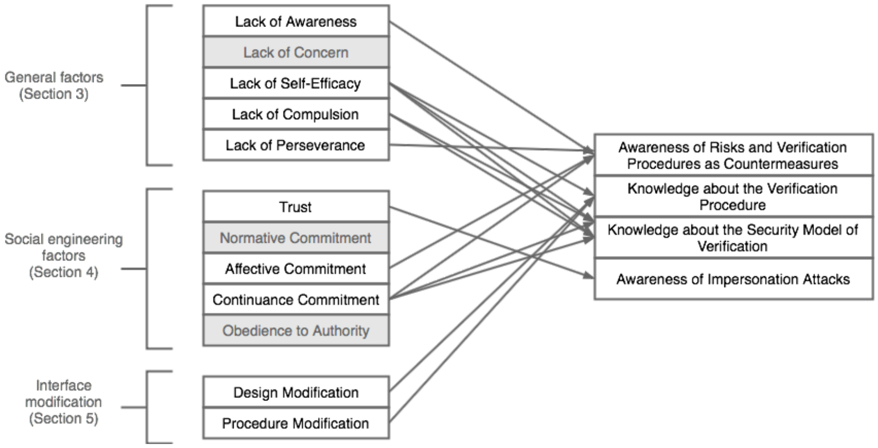


Fig. 1. The countermeasures addressing the factors outlined in Sections 3 to 5. The factors with questionable relevance for the electronic voting context are greyed out.

without making voters taking care of its security. Thus, it might also be necessary to make voters aware that it is impossible to have a voting system, including paper-based voting, which is totally free of the vote manipulation risk. On the other hand, verifiability has been identified as a measure to increase trust in the voting system in previous research [25, 28]. Furthermore, empiricial studies show that the voters who are concerned about the security of electronic voting would be more willing to trust the system if it provides verifiability possibilities such as personalised codes on each ballot (note that other commonly used approaches for cast-as-intended verifiability were not mentioned in the study) [17]. Hence, further investigations on the reactions of voters once they are made aware of risks of manipulating votes and corresponding verifiability countermeasures are needed.

The knowledge of security provided by the verification can furthermore be helpful to offset the potential usability problems of the verification. In the current state of research, verification procedure requires extra steps from the voter. As discussed in previous chapters, this additional effort becomes a danger if it prevents the voters from verifying, either because they are generally unwilling to dedicate too much effort (Section 3.4), or actively discouraged by the adversary to do so (Section 4.3). While the time and effort required for the verification can sometimes be minimized via usability improvements, often the additional steps in the verification are are inevitable in order to ensure the security of the verification. As mentioned in Section 3.4, previous studies show that the voters are ready to accept the additional effort if they understand the security benefits it brings. Hence, while generally the verification processes should be designed to be as efficient as possible, an appropriate trade-off with security should be carefully considered and communicated to the voters.

Furthermore, as discussed in Section 3.5, the perception of the society, that the verification is unnecessary unless one is particularly concerned about the risks of vote manipulation, can hinder the voters' readiness to verify. As discussed in Section 4.4, the adversary can exploit such a societal attitude by persuading the voter, that as long as they are willing to trust the government or the groups in favor of introducing electronic voting, they should not verify. It is therefore important to ensure, that the voters understand the general importance of verification as a stepstone into ensuring the integrity of democratic institutions without perceiving the need to verify as mistrust in the institutions.

Educating about Procedure. Once they are aware of the need of verifiability and the need for them to take actions it is necessary to explain the procedure to them (in order to increase the level of self-efficacy, see Section 3.3. However, they should know that one possible adversary strategy is to modify the interfaces to make it less likely that voters verify (Section 5). They should know whom to contact in case they detect a modification.

Explaining Security Model. The voter's lack of knowledge about the security that the verification provides can furthermore hinder them from verifying, if they believe that the verification is either futile or dangerous (see Section 4.3). Hence, education measures are needed that explain the security model of the verification and address potential misconceptions .

Raise Awareness of Impersonation Attacks. It is furthermore important to make voters aware of possible social engineering attacks that involve the adversary impersonating a trustworthy entity to the voter (see Section 4.1). As such, the voter should be able to detect whether their communication with the voting system is genuine. If the voter have the option to select a trusted third party to perform the verification on their behalf, the trustworthiness of such a party should be clearly communicated to them, ideally with an option to validate it from an independent source. For this, further research into trust communication is required.

6.2 Future Work

While ensuring cast-as-intended verifiability is a crucial step towards the security of electronic voting, it is not enough to prevent election manipulation on its own. As such, measures towards protecting against server-side attacks have to be implemented, which is, however, out of scope of this work.

The factors and countermeasures outlined in the paper focus on the voters who are generally willing to follow the voting protocol, or at least do not actively try to violate it. Hence, we did not consider the issue of vote buying, where the adversary does not try to deceive a law-abiding voter, but the voter willingly collaborates with the adversary instead. As the issue of vote buying is crucial in electronic voting, particularly, in remote voting, we consider the consideration of vote buying from a human-centered perspective an important part of future work.

As consider the cast-as-intended verifiability in Internet voting, while some of our results are likely to be transferred to other channels of electronic voting, the specific scenarios, such as polling-place voting machines, remain the topic for the future work. Furthermore, as the security of electronic voting and paper-based voting (polling place or postal) has been the topic of previous research [12, 30], it would also be possible to compare the security issues related to human factor and voter verifiability between these two voting channels.

Acknowledgment

This work has been supported by the German Federal Ministry of Education and Research within the Competence Center for Applied Security Technology (KASTEL). It has also been supported within the project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 740923, project GHOST (Safe-Guarding Home IoT Environments with Personalised Real-time Risk Control).

References

1. Acemyan, C.Z., Kortum, P., Byrne, M.D., Wallach, D.S.: Usability of voter verifiable, end-to-end voting systems: Baseline data for helios, prêt à voter, and scantegrity ii. The USENIX Journal of Election Technology and Systems 2(3), 26–56 (2014)
2. Adida, B.: Helios: Web-based open-audit voting. In: USENIX security symposium. vol. 17, pp. 335–348. USENIX Association (2008)
3. Alkaldi, N., Renaud, K.: Why do people adopt, or reject, smartphone password managers? In: 1st European Workshop on Usable Security (EuroUSEC) (2016)
4. Brightwell, I., Cucurull, J., Galindo, D., Guasch, S.: An overview of the ivote 2015 voting system. Tech. rep. New South Wales Electoral Commission (2015)
5. Budurushi, J., Neumann, S., Olembo, M.M., Volkamer, M.: Pretty understandable democracy-a secure and understandable internet voting scheme. In: Availability, Reliability and Security (ARES), 2013 Eighth International Conference on. pp. 198–207. IEEE (2013)
6. Fuglerud, K.S., Røssvoll, T.H.: An evaluation of web-based voting usability and accessibility. Universal Access in the Information Society 11(4), 359–373 (2012)
7. Galindo, D., Guasch, S., Puiggali, J.: 2015 neuchâtel's cast-as-intended verification mechanism. In: VoteID 2015: 5th International Conference on E-Voting and Identity. pp. 3–18. Springer (Sep 2015)
8. Guasch Castelló, S.: Individual Verifiability in Electronic Voting. Ph.D. thesis, Universitat Politécnica de Catalunya (2016)
9. Heiberg, S., Martens, T., Vinkel, P., Willemson, J.: Improving the verifiability of the estonian internet voting scheme. In: International Joint Conference on Electronic Voting. pp. 92–107. Springer (2016)
10. Heiberg, S., Parsovs, A., Willemson, J.: Log analysis of estonian internet voting 2013–2014. In: International Conference on E-Voting and Identity. pp. 19–34. Springer (2015)

11. Karayumak, F., Olembo, M.M., Kauer, M., Volkamer, M.: Usability analysis of helios-an open source verifiable remote electronic voting system. Electronic Voting Technology Workshop/Workshop on Trustworthy Elections EVT/WOTE '11 (2011)
12. Krimmer, R., Volkamer, M.: Bits or paper? comparing remote electronic voting to postal voting. In: EGOV (Workshops and Posters). pp. 225–232 (2005)
13. Kulyk, O., Neumann, S., Budurushi, J., Volkamer, M.: Nothing comes for free: How much usability can you sacrifice for security? IEEE Security & Privacy 15(3), 24–29 (2017)
14. tom Markotten, D.G.: User-centered security engineering. In: Proceedings of the 4th EurOpen/USENIX Conference–NordU2002 (2002)
15. Marky, K., Kulyk, O., Renaud, K., Volkamer, M.: What did i really vote for? In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. p. 176. ACM (2018)
16. Marky, K., Kulyk, O., Volkamer, M.: Comparative usability evaluation of cast-as-intended verification approaches in internet voting. In: SICHERHEIT 2018. pp. 197–208. Gesellschaft für Informatik e.V. (2018)
17. Milic, T., McArdle, M., Serdült, U.: Haltungen und bedürfnisse der schweizer bevölkerung zu e-voting. Tech. rep., Aarau: Zentrum für Demokratie Aarau (2016), https://doi.org/10.5167/uzh-127938
18. Nemeslaki, A., Aranyossy, M., Sasvári, P.: Could on-line voting boost desire to vote?–technology acceptance perceptions of young hungarian citizens. Government Information Quarterly 33(4), 705–714 (2016)
19. Neumann, S., Olembo, M.M., Renaud, K., Volkamer, M.: Helios verification: To alleviate, or to nominate: Is that the question, or shall we have both? In: International Conference on Electronic Government and the Information Systems Perspective. pp. 246–260. Springer (2014)
20. Nielsen, J.: Enhancing the explanatory power of usability heuristics. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems. pp. 152–158. ACM (1994)
21. Olembo, M.M., Renaud, K., Bartsch, S., Volkamer, M.: Voter, what message will motivate you to verify your vote. In: Workshop on Usable Security, USEC (2014)
22. Olembo, M.M., Bartsch, S., Volkamer, M.: Mental models of verifiability in voting. In: International Conference on E-Voting and Identity. pp. 142–155. Springer (2013)
23. Schneider, S., Llewellyn, M., Culnane, C., Heather, J., Srinivasan, S., Xia, Z.: Focus group views on pret a voter 1.0. In: Requirements Engineering for Electronic Voting Systems (REVOTE), 2011 International Workshop on. pp. 56–65. IEEE (2011)
24. Simpson, R., Storer, T.: Third-party verifiable voting systems: Addressing motivation and incentives in e-voting. Journal of Information Security and Applications (2017)
25. Spycher, O., Volkamer, M., Koenig, R.: Transparency and technical measures to establish trust in norwegian internet voting. In: International Conference on E-Voting and Identity. pp. 19–35. Springer (2011)
26. Volkamer, M., Alkassar, A., Sadeghi, A.R., Schulz, S.: Enabling the application of open systems like pcs for online voting. In: Proc. of Workshop on Frontiers in Electronic Elections (2006)
27. Volkamer, M., Renaud, K., Kulyk, O., Emeröz, S.: A socio-technical investigation into smartphone security. In: International Workshop on Security and Trust Management. pp. 265–273. Springer (2015)
28. Volkamer, M., Spycher, O., Dubuis, E.: Measures to establish trust in internet voting. In: Proceedings of the 5th International Conference on Theory and Practice of Electronic Governance. pp. 1–10. ACM (2011)

29. Weber, J.L., Hengartner, U.: Usability study of the open audit voting system helios. http://www.jannaweber.com/wpcontent/ uploads/2009/09/858Helios.pdf (2009)
30. Willemson, J.: Bits or paper: Which should get to carry your vote? Journal of Information Security and Applications 38, 124–131 (2018)
31. Workman, M.: Gaining access with social engineering: An empirical study of the threat. Information Systems Security 16(6), 315–331 (2007)
32. Workman, M.: Wisecrackers: A theory-grounded investigation of phishing and pretext social engineering threats to information security. Journal of the Association for Information Science and Technology 59(4), 662–674 (2008)
33. Zetter, K.: The myth of the hacker-proof voting machine. https://www.nytimes.com/2018/02/21/magazine/the-myth-of-the-hacker-proof-voting-machine.html (2018), online; accessed: 15-May-2018

Defining a national framework for online voting and meeting its requirements: the Swiss experience

Jordi Puiggalí[1[0000-0003-1472-415X]] and Adrià Rodríguez-Pérez[1[0000-0002-5581-1340]]

[1] Scytl Secure Electronic Voting, S.A.
Research and Security Department
08008 Barcelona, Spain
{jordi.puiggali, adria.rodriguez}@scytl.com

Abstract. Switzerland is a worldwide reference for direct democracy. In 2000, the country started piloting internet voting as a complementary voting channel. This introduction followed a security-before-speed approach, starting with three cantons only and a limitation on the electorate that could use the system in federal elections and votes. Gradual cantonal adoption was carried out after analysing experiences' results, and by 2015 more than half of the cantons were offering internet voting. In 2014 Switzerland made a major step in its online voting strategy by enforcing a national internet voting regulation that enables cantons to increase the electorate thresholds for federal elections. The regulation introduces three different security levels linked to an increase of the former cantonal electorate limit from 30% to 50% or even 100%. For each level, the regulation defines specific security requirements to be achieved, as well as verifiability mechanisms (individual and universal) for the two top levels. The regulation also describes how to certify these verifiability mechanisms, introducing the need for security and symbolic proofs to evaluate the design of the cryptographic protocols (in addition to security evaluation standard frameworks like Common Criteria or ISO 27001). In this paper, we describe the regulation framework, analyse how it has been implemented in the four years following its adoption and illustrate how it has influenced the evolution plans of the different voting systems. We conclude by identifying some of the benefits of this framework and by describing the future plans that the government is considering for its online voting strategy.

Keywords: Switzerland, internet voting, regulation, verifiability, certification, online voting experiences.

1 Introduction

Switzerland has been one of the main references on the introduction of online voting at a national level. It is one of the first countries that introduced internet voting for politically binding elections[1] [1, 2], first by allowing its use in 2000 and following with a

[1] Previous experiences include the UK internet voting pilots during the local elections of May 2002. At the same time, it is reported that the first binding internet election to take place

limited set of pilots by three cantons [3]. The first binding processes using internet voting were carried out in 2003 (in the Canton of Geneva) and 2005 (in Neuchâtel and Zurich). The approach followed by Switzerland was to prioritise the security versus implementation speed, so the gradual introduction of internet voting has always been based on minimising the risks and analysing the maturity of technology before going a step further and facilitating its adoption by a wider electorate. Based on this strategy, in 2014 Switzerland became once again a new reference when introducing the first regulation for online voting systems that included advanced security requirements such as verifiability and a mechanism for certifying them (e.g. cryptographic and formal proofs). The regulation establishes three security levels that limit the percentage of the electorate that can use internet voting for federal elections (i.e. accepted risk): 30% of the cantonal electorate at the first level, 50% at the second and 100% (i.e. no limit) for the third. Depending on the verifiability mechanisms implemented by the voting systems, the two higher levels can be achieved when individual verifiability is provided (level 2) or complete verifiability, namely: individual and universal (level 3).

Unlike other countries were internet voting is being used or has been piloted, Switzerland is the only country that has developed an overarching certification and authorisation framework suitable to evaluate different internet voting systems, both from an operational and technical perspective. This allows to certify more than one voting system technologies if they achieve the security requirements, so each canton can choose which one they prefer based on this certification. Other countries, like Estonia, Norway or France mainly use a public tender process to choose a vendor for developing the requirements of the voting systems[2].

Thus, the goal of this paper is to carefully evaluate the new Swiss internet voting framework. To do so, the paper starts by analysing the evolution of the Swiss internet voting regulation from its introduction until its update in 2014. This will allow us to identify the steps followed by the Swiss government before adopting the new regulation. In the second section of the paper, we describe the new regulation in detail as well as the evaluation requirements for authorising the use of voting systems in place since 2014. Following, section 3 continues explaining the impact of the application of the regulation during the four years in which it has been in force, especially in the evolution of the voting systems' technologies and cantons' online voting landscape. Finally, in the conclusions, we address the future plans by the Swiss government regarding its online voting strategy and try to draw some lessons and best practices that the authors have detected from the implementation of the regulation.

occurred in the Unites States during March 2000, in the framework of the Arizona Democratic Party primary elections.

[2] Only the US has a certification framework for voting systems since 2002 (currently known as Voluntary Voting System Guidelines). The difference between the Guidelines and the Swiss framework is that the former only applies to poll site voting systems and its adoption by the States is voluntary. On the other hand, Swiss cantons cannot use in a federal election or referendum an internet voting system that has not been authorised by the Federal Chancellery).

2 Switzerland: the long road towards complete verifiability

2.1 A tradition of direct democracy

Switzerland has a long history of direct citizen participation in decision-making processes, with roots that start in the 13th century. In Switzerland there are popular votes at least four times a year on the national, cantonal and communal levels [4]. Besides elections where voters choose their representatives (e.g. at the communal, cantonal and federal levels), citizens can participate in several other voting events. Citizens can also propose popular voting initiatives on their own (after having obtained enough popular support by collecting signatures). These proposals can be voted in referendums in which citizens are asked to express their opinion. Proposals can include a new law or an amendment of the Constitution, among others.

At the same time, the introduction of internet voting in Switzerland is much linked to the country's federalism [5]. In Switzerland, the implementation of elections and referendums are a subnational matter. While there is an overarching umbrella legislation at the country level, which is aimed at ensuring citizen's political rights, it is the cantons (and in some cases even the local executives, the communes) who maintain vote registries, organise elections and determine voting results. Therefore, the action by the Federal Council (highest national executive power in Switzerland) is rather limited as the authority approving internet voting trials and formulating the specific conditions under which the new digital channel can be implemented (Art. 8a FAPR) [6]. There are detailed provisions on prerequisites for internet voting trials put forward at the federal level. Cantons are, however, completely free to decide whether they want to offer internet voting or not [7].

2.2 The Gradual introduction of internet voting

In Switzerland, postal voting has been reported to be used in the canton of St. Gallen as early as of 1673 [8]. The postal vote procedure was introduced by the federal government and most cantons in the early 1990s. In 2006, all Swiss cantons were offering postal voting as an alternative channel for citizens to cast their votes in advance, and studies show that up to 81,5% of voters use postal voting as their preferred channel [9]. The high ratio of use of advance voting has been said to go along many deep changes in voting behaviour, changes usually associated with internet voting [10].

In 1998, the Federal Council proposed to study the introduction of internet voting and in 2000 the Swiss Parliament asked the Federal Council to start a programme for actively piloting electronic voting in Switzerland. A study on the feasibility and risks of internet voting was initiated by the Federal Council with the principle of prioritising security in front of wide adoption speed [7]. Based on this principle, binding internet voting pilots were proposed in 2002 to cantons that volunteered to participate. The objective was to evaluate the results of these pilots in 2005 before deciding whether to open the use of internet voting to the whole Federation. Three cantons participated in these pilots: Geneva, Neuchâtel, and Zurich.

Geneva designed and developed its own voting system in partnership with two external companies (Hewlett Packard and Wisekey). The ownership of the system and its operation was kept by the State Chancellery of Geneva, that assumed also the hosting, management and evolution of the voting system. The system based its security on using standard security mechanisms, such as the encryption of the communications using SSL and encryption of the ballots in the voting server using standard cryptographic algorithms. The system was used for the first time in a referendum in 2003.

Neuchâtel acquired its online voting technology from an existing provider (Scytl Secure Electronic Voting, S.A.) and integrated it into its e-government portal (*Guichet Sécurisé Unique*). This portal offers different online services to citizens, including online voting. Voters need to register as users in this portal to get access to the voting system. The voting system provided encryption and a digital signature of the votes in the same voting terminal (Java applet), anonymous decryption based on a mix-net and recorded-as-cast verifiability through voter receipts. The first referendum using internet voting carried out in Neuchatel was organised in 2005.

Zurich hired the services of Unisys to implement and manage its online voting system. Zurich's system main security characteristic was the use of codes to select the candidates: voters received a special voting card with a unique code per candidate and had to type the code of their preferred candidate instead of selecting them. This mechanism preserves the privacy of the voters even if they are using an insecure communication channel (e.g., SMS). The voting system was used for the first time in 2005.

In 2006 the three pilots were successfully evaluated, and the Federal Government opened the door for all the cantons to use internet voting. Yet, a restriction was set: the internet voting system could only be used by up to 20% of the cantonal electoral roll (later extended to 30%), and by up to 10% of the Swiss electoral roll. The purpose of this limitation was to reduce the impact of any early adoption problem while the systems were being introduced. Regarding security evaluation, the Federal Chancellery (the staff organisation of the Swiss federal government) did not define a specific framework and, therefore, this was mainly delegated to the cantons' discretion.

After opening the use of internet voting to all cantons, Geneva and Zurich started to offer their voting system to other cantons. In 2009, the cantons of Aargau, Fribourg, Graubünden, Schaffhausen, St. Gallen, Solothurn, and Thurgau created the e-voting "Consortium", using Zurich's e-voting system. On their side, the cantons of Basel-Stadt, Lucerne, and Bern reached an agreement with Geneva to use its system. At the same time, Geneva's system was also updated to improve its security, including a Java applet that encrypted the votes already in the voters' device and a mixing process that anonymised all ballots before decryption. By 2015, more than half of the Swiss cantons were offering internet voting to their citizens.

It is also worth mentioning that during this time there were also some drawbacks to the introduction of internet voting in Switzerland. In Geneva, for instance, the piloting of internet voting was stopped during 2005-2007 since opponents claimed that without a proper legal basis this voting channel should not be used [11]. Shortly after, however, the majority of voters in Geneva backed the introduction of internet voting as an official channel into their Constitution, with an overwhelming approval rate of 70%. In the canton of Zurich, the internet voting project was halted for technical reasons in 2011 [12]. While at the beginning this was only thought to be temporary, Zurich actually never reintroduced the possibility for its citizens to vote online.

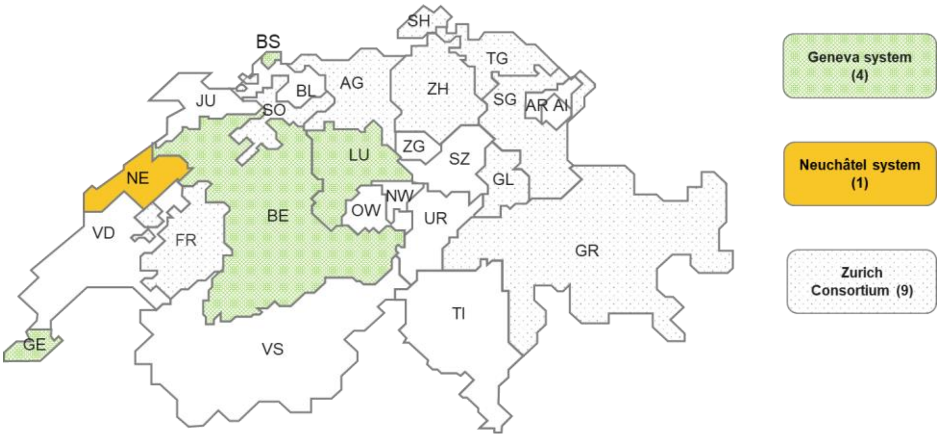


Fig. 1. Status of internet voting adoption in Switzerland (2006-2013)

3 The latest regulation from 2014

3.1 The authorisation process

In 2011, the Federal Council of Switzerland set up a task force to study the security issues of internet voting, with the purpose of potentially expanding and setting up a framework for evaluating the security of the existing voting systems. Since the introduction of internet voting, the security requirements of the voting systems had been decided by each canton under their own security practice rules. The work of the task force allowed for the set-up of a new regulation for internet voting at the country level. This new regulation, which became binding in January 2014, amended the Ordinance on Political Rights of 24 May 1978 (OPR) in those sections addressing the authorisation for the conduct of trials with electronic voting, as well as the requirements that the systems had to meet for their use by the citizens [13]. The main innovations of the new regulation were twofold.

On one side, the Swiss Federal Council set up new limits for the use of internet voting by the cantons (Art. 27f. OPR), allowing for the participation of a maximum of the electorate as follows:

- A maximum of 30% of the cantonal electorate[3]; at the same time with a limit of 10% of the entire Swiss electorate.

[3] According to Art. 27f.2 OPR, expatriate Swiss citizens who are eligible to vote are not included when calculating the limits.

- A maximum of 50% of the cantonal electorate; at the same time with a limit of 30% of the entire Swiss electorate;
- The entire cantonal electorate.

The amendment of the OPR also established that it should be the Federal Chancellery who stipulates the requirements that an electronic vote casting system and its operation must meet at each level[4]. As a result, the Federal Chancellery published an Ordinance on Electronic Voting (VEleS), specifying the requirements for authorizing electronic voting for each of the new limits [14]. These requirements do not cover only functional, usability requirements and general security aspects, but they also follow the path initiated by Norway by requiring state-of-the-art-verifiability functionalities (cast-as-intended, recorded-as-cast and counted-as-recorded verifiability)[5].

On the other side, the Amendment of December 13 also set up clear procedures for cantons to be authorised to use internet voting. According to the new regulation, trials on electronic vote casting in federal popular votes require an initial licence from the Federal Council (Art. 27a.1 OPR). The regulation sets that only after at least five successive problem-free individual trials in a canton in federal ballots may the Federal Council permit this canton to use electronic vote casting in federal popular votes (Art. 27a.3 OPR). Cantons who have received an initial licence must place an authorisation request for electronic vote casting in front of the Federal Chancellery prior to every voting process (Art. 27e.1 OPR). According to the regulation, the authorisation shall be granted if the system and the operational modalities (limits) chosen by the canton meet the requirements specified by the Federal Chancellery.

Beyond the participation of the applicant canton, the Federal Council, the Federal Chancellery, and several agents are involved in the licensing and authorisation processes. On the one side, the regulation requires an independent, external agency to (a) confirm that the Federal Chancellery's security standards are met; and (b) to review whether the safety precautions and the electronic vote casting system complies with the state of the art (Art. 27l.1 OPR). This independent external agency has taken the shape, on many occasion, of what has been called as *groupes d'accompagnement* (which could translate from French as steering, support or advisory groups) [17]. In practice, *groupes d'accompagnement* are made up by representatives from four different cantons using a different voting system. In case that a license is requested for the deployment of an internet voting system for less than 30% limit of the electorate in a canton, a *group d'accompagnement* is set up to validate the functioning of the system and its deployment (Art. 511 of the Catalogue). In case that the licence concerns more than the 30% of the cantonal electorate limit (i.e. up to 50% or the whole cantonal electorate), more

[4] The Amendment of 13 December 2013 to the OPR establishes that "Cantons that involve their entire electorate in a trial must guarantee that the correct procedure for electronic vote casting and the accuracy of the results from this vote casting system can be verified" (Art. 27i.2 OPR). Following, the Federal Chancellery has regulated the specific verifiability requirements for the deployment of internet voting in the different categories set above (Annex 4 to VEleS).

[5] Due to space limitations, we will not be able to provide definitions for most of the technical concepts used in the paper. In this regard, for a better understanding of verifiability concepts we recommend the reading of [15], and of [16] on practical experiences.

specific examinations are set (Annex 5 to the VEleS). In this second scenario, the participation of specialized institutions is required, including the participation of institutions accredited by the Swiss Accreditation Service (SAS) or certification agencies, among others. These specialized institutions (i.e. certification laboratories), should first pass an accreditation process through the Swiss Chancellery to get the seal of certification authorities of the VEleS regulation.

Finally, academic support may be also provided with a view to ensure that trials with electronic vote casting are investigated as to their effectiveness. In particular, the trends in voter turnout and the effects over the vote casting habits (Art. 27o.2 OPR).

More recently, the Federal Chancellery set up an expert group to start working on the full dematerialization of the election, namely: the complete digitisation of the voting process [18]. With a total of 13 members, the group gathers representatives from Academia, the Confederation, the cantons, as well as from internet voting providers. In April 2018, the group of experts issued its final report [19], which was made public in late June. In the meantime, VEleS was also amended. As of July 2018, the Ordinance requires certification of internet voting solutions at all levels, provided that they offer complete verifiability. The amendment also introduces a new requirement for the certification of these solutions, namely: the publication of its source code.

In what follows we will describe the requirements of the legislation approved as of 2014, including the recent amendments to VEleS.

3.2 Different levels of authorisation

The definition of the requirements in three different levels was intended for a gradual implementation of internet voting in the cantons, with security in mind: systems authorised for the basic level (level 1) can be used by up to 30% of the cantonal electorate, as far as they fulfil a set of functional and security requirements. Stronger requirements, mainly in verifiability as well as in testing and certification, are required for systems that want to reach larger parts of the electorate, such as 50% of the cantonal electorate (level 2) or the whole cantonal electorate (level 3).

Security requirements. In contrast to the previous regulation, which left the decision of the system's security functionalities mostly to the cantons, the new Swiss regulation defines a standard set of security requirements taken in part from the BSI Common Criteria Protection Profile for Internet Voting Products [20] and the standards of the Council of Europe [21, 22, 23]. The regulation also provides an analysis of general threats to electronic voting systems, security objectives (based on those from Common Criteria) and functional requirements of the system.

In order to be authorised at any level, systems have to provide a risk assessment which shows how the security requirements are implemented, the risks mitigated, and the security objectives met. Experts at the Federal Chancellery then evaluate the documentation provided to decide whether the voting system is compliant with the expected security requirements.

Verifiability requirements. These requirements were introduced for level 2 and level 3 authorisation. The Swiss regulation classifies the verifiability properties in two main categories: individual verifiability (covers what is known as cast-as-intended and recorded-as-cast verifiability) and universal verifiability (counted-as-recorded verifiability). These requirements also include the concept of complete verifiability, equivalent to the end-to-end verifiability concept, which is the result of the union of both individual and universal verifiability properties as described in the regulation. The regulation requires individual verifiability for systems at the authorisation level 2 (maximum of 50% of the electorate at cantonal level and 30% of the entire Swiss electorate), and complete verifiability for systems at the authorisation level 3 (the entire electorate).

In both cases, there are different trust assumptions regarding the entities that participate in the verification protocol. In authorisation levels 2 and 3, neither the voter computer nor the vote transmission entities are trusted. However, for level 2 authorisation the server-side entity participating in the generation of the individual verifiable proof of content (e.g., generation of Return Codes[6]) can be trusted. While in level 3 the server-side entity is not trusted anymore, neither for generating such proof of content (Return Codes), nor for ensuring the storage of the votes in the Ballot Box (recorded-as-cast verification). Instead, at level 3 the regulation requires using a set of Control Components for generating the proof of content. These Control Components interact with the voting system and are in charge of implementing sensitive operations in a distributed way. Therefore, all the Control Components have to agree on the validity of the contents of a ballot by providing a proof of the cryptographic operations that they perform. Furthermore, all of them store a copy or a fingerprint of the encrypted votes that they have processed in order to ensure the integrity of the Ballot Box.

Additionally, Control Components in level 3 are also in charge of the protection of voters' privacy. In the counting process, this is usually implemented by using a mix-net with multiple nodes to anonymise the votes before decryption, where each node is run by a different Control Component (properties of mix-nets guarantee that, as far as one of the mix-nodes is honest, voters' privacy is preserved)[7].

In order to respond to the necessarily high trust assumptions put on them, Control Components have to fulfil a set of requirements to guarantee their independence in terms of operation, infrastructure, software, and monitoring. The Swiss requirements for authorisation at level 3 are the first to require the whole distribution of sensitive operations into independent components for real-world systems.

Certification and testing requirements. The original version of VEleS did not require to go through a certification process by an accredited entity for authorisation at level 1. However, VEleS was amended in June 2018 to mainstream certification requirements at all levels. Since July 2018, these certification requirements are now compulsory regardless of the percentage of the cantonal electorate which is going to be authorised to vote online, provided that the system used offers complete verifiability[8].

Before the amendment, the testing and evaluation of the system at level 1 were mainly carried out by the experts of the Chancellery and the *group d'accompagnement*.

[6] To better understand how Return Codes work, the reading of [24] is recommended.

[7] On mix-nets and other anonymisation mechanisms, the reading of [25] is suggested.

[8] Since the current paper has been written by the time the 2018 amendment has entered into force, we address both the previous and the current requirements in this section.

Testing and evaluation were based on examining the voting system documentation (e.g. threat assessment) as well as a live testing of the voting system. As for levels 2 and 3, the voting system needed to be certified by an accredited entity before being tested by the experts of the Chancellery and the *group d'accompagnement*.

Both before and after the amendment, the certification process had the following main objectives:

- Certify the cryptographic protocol verifiability's compliance: The design of the cryptographic protocol is evaluated to check its compliance against the verifiability requirements defined in the regulation (individual verifiability for level 2 and complete one for level 3). To this end, the regulation requires the implementation of security (cryptographic) and symbolic (formal) proofs of the protocol that implements the verifiability mechanisms. Cryptographic experts from the certification authority will analyse the correctness of both proofs and evaluate the conformity of the trust assumptions with the regulation (e.g., the need that the client side is not trusted to achieve the properties). These proofs are also cross-checked with the protocol's design documentation to ensure that the protocol implemented in the voting system is aligned with them.
- Certify the software security and functionality: The software design and security functional requirements are also evaluated by the certification entity. To this end, the regulation requires to use as reference the Common Criteria standard over the BSI Protection Profile for Internet Voting systems. For level 2 certification, it is required to use the processes for assurance level 2 (EAL2) while for level 3 certification it requires to use the same assurance level 2 except for the Control Components, that require assurance level 4 (EAL4). For testing the security of the software, the regulation recommends using OWASP Top 10 testing guidelines.
- Certify the security of the infrastructure and its resilience against intrusions: the certification authority also reviews the security controls implemented in the deployment of the voting system and its infrastructure. To this end, ISO/IEC 27001:2005 is used as a reference.
- Certify the requirements for printing offices.

The main novelty since the amendment has entered in force is that certification at level 1 is also required for systems providing complete verifiability. There are, however, some differences for certification at level 1 if we compare its requirements to certification for levels 2 and 3:

- Regarding the functionality, it is not necessary to certify the software of e-administration portals which may be linked to the online voting solutions.
- At the infrastructure and exploitation level, the certification can be limited to the infrastructure which is used to store the ballots and to provide for the properties of individual verifiability (certification level 2).
- The requirements for the printing offices are out of the scope in this certification.

4 Swiss internet voting authorisation process in practice

When the new regulation came into force in January 2014, all the pre-existing voting systems were called for certification once again, to allow their use in the 2015 elections. Thus, all cantons using internet voting started the process to be authorised against the basic level (level 1) to keep at least the same status. Geneva and Neuchâtel also took advantage of this opportunity to improve their voting systems and prepared them for achieving higher levels of certification. In parallel, the Federal Chancellery started the accreditation process for external entities required for the certification levels 2 and 3.

4.1 Level 1 authorisation: the loss of the Consortium

During 2015, the three voting systems already in use in Switzerland (Geneva, Neuchâtel and the Consortium) started at level 1 authorisation process [26]. Geneva and Neuchâtel started this process with a new version of the voting system, bearing in mind higher level requirements of the regulation (e.g. individual verifiability mechanisms) [16]. Cantons in the Consortium did not make any changes to their voting system.

The requirements for authorising a voting system at level 1 use as reference a risk assessment conducted by the experts of the Chancellery. This allows to evaluate how the security measures implemented by a voting system mitigated existing risks. The experts of the Chancellery also evaluate the security design and practices for the implementation of the voting system (e.g. OWASP). The authorisation process ends with a security and functional testing and an evaluation of a live version of the voting system by the experts of the Chancellery and the members of the *group d'accompagnement*.

Neuchâtel and Geneva got authorised with the evolution of their previous voting systems. However, because of the increase in security requirements in the new regulation, the voting system of the Consortium lost its previous authorisation since it failed to address the security issues detected by the experts of the Chancellery. The main reason was the approach followed by the Consortium when implementing their online voting solution: they hired a generic consultancy firm (Unisys) to develop and support it. Therefore, any change to the voting system required to subcontract an external entity to do it. Geneva had its own development team and Neuchâtel was using the pre-existing software of an online voting system provider (Scytl), who helped them adapt the voting system to the new requirements.

The de-authorisation of the Consortium's voting system forced the dissolution of the Consortium, leaving 9 cantons without internet voting and reducing the total number of authorised cantons to 5.

In parallel, in 2015 Swiss Post made a partnership with Scytl to develop an online voting system that could be offered to any Swiss canton. This opened the door for cantons who belonged to the Consortium to choose between Geneva and Swiss Post's voting systems should they wish to continue offering online voting to their citizens. Neuchâtel never offered its voting system to other cantons.

In 2016, the Canton of Fribourg chose Swiss Post's voting system and achieved the level 1 authorisation, becoming the first reference for Swiss Post's voting system. Also, in 2016 Neuchâtel decided to move its voting system to Swiss Post's, achieving level 1 authorisation in 2017.

During 2017, Aargau and St. Gallen also achieved level 1 authorisation with Geneva's voting system, while Thurgau and Basel Stadt (currently using Geneva's voting system) decided to adopt Swiss Post's voting system. Their authorisations are expected during 2018 and 2019 respectively.

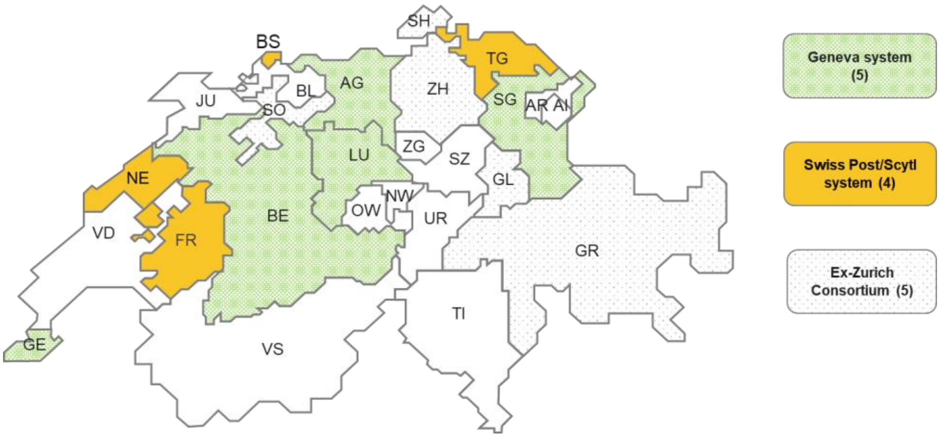


Fig. 2. Status of internet voting adoption in Switzerland (2018)

4.2 Level 2 authorisation: the first certified voting system

In 2016, Swiss Post started the process with the Federal Chancellery to have its voting system certified at level 2. This was an important step in Switzerland since it was the first solution certified at this level and opened the door for cantons to increase the use of internet voting for up to 50% of its electorate in federal elections and votes. At first, this may not seem relevant, since as of March 2018 only an average[9] of 25,78% of the voters authorised to use the electronic channel ended up casting their votes. One could understand that these data do not seem to support the idea that extending the franchise may be required in the short them. At the same time, however, it is worth pointing out that internet voting is becoming nowadays one of the main channels used by voters. For instance, these same data show that in March 2018 the electronic voting channel was used by 49,08% of voters on average, and in some cases by up to 64,69% of voters [27]. Taking account of current limits, if higher percentages of the electorate wish to cast their vote electronically, cantons will only be able to extend the franchise to more than 30% of its resident cantonal electorate if they use Swiss Post's solution.

[9] for the eight cantons offering internet voting: Bern, Lucerne, Fribourg, Basel-Stadt, Saint Gallen, Argovie, Neuchâtel and Geneva.

Also, in 2016 the Chancellery received the application from KMPG to start the accreditation process to become a VEleS certification authority. Therefore, KPMG accreditation process was done in parallel to the certification process of Swiss Post's voting system.

In 2017, Swiss Post had all the information ready to start the certification process at level 2. This included:

- The security [28] and symbolic [29] proofs of the cryptographic protocol implementing the Return Codes for individual verifiable mechanism, following the logic developed for Neuchatel's system in 2015 [24]. These proofs were based on the assumption that the voting client and the server-side's Return Codes generation processes do not collude.
- All the documentation required by the Common Criteria framework for certifying the solution at assurance level 2 (EAL2) with the Protection Profile published by the BSI. The documentation included the Target of Evaluation, the security design, the threat assessment, the traceability of the security functional requirements and any other evidence required by the framework.
- All the documentation about the system architecture and security controls required to validate the compliance with ISO 27001.

The certification process started in early 2017 and, after several iterations, the certification was achieved in September 2017. One of the more complex parts of the certification process was the evaluation of the security and symbolic proofs. The lack of experts in this area (mainly an academic discipline) was solved by KPMG by involving the ETH Zurich in the evaluation process.

Nowadays, Swiss Post is working in the authorisation at level 2 for Thurgau Canton, which is expected to take advantage of it by September 2018.

4.3 Level 3 authorisation: objective 2019 elections

In 2016, while Swiss Post had started working in achieving certification at level 2, Geneva announced that they would completely re-design their system and build a new one with the goal of achieving certification at level 3 in two years (2018) [30]. To this end, Geneva closed a collaboration agreement with the Bern University of Applied Sciences (BfH) for the design of this new voting system. A new protocol was designed by the BfH experts in 2016 [31] and they implemented a proof of concept in 2017, at the same time that their source code was made public [32]. The source code made available was not a fully functional voting system but an E2E validator of the protocol. In 2017 some changes in the protocol were made to solve some detected attacks [33]. BfH is also working with external experts to implement the security and symbolic proofs of the protocol. The goal of Geneva is to achieve the certification to allow cantons to be authorised at level 3 for the next federal elections in 2019.

In the case of Swiss Post, they also have plans to achieve the certification at level 3 ahead of the 2019 federal elections. Having already achieved certification at level 2 means that part of the certification effort (that of individual verifiability requirements) is already achieved. This will therefore allow them to focus on the requirements dealing with complete verifiability (i.e. the Control Components).

4.4 Future plans

While Swiss Post and Geneva were working on achieving the new certification levels, the Federal Chancellery and the Federal Council defined a new instrument for planning the introduction of internet voting [34, 35]. This planning sets six objectives, which include the dematerialisation of the vote and measures to improve the confidence in the online voting systems. To address these issues, the Federal Council established a group of experts on internet voting in August 2017 [18].

Regarding the certification and authorisation processes described in this paper, and with a view to making internet voting the third ordinary voting channel country-wide, the expert group has recommended maintaining the current security requirements, with a special focus on traceability and confidentiality, accessibility, and transparency. At the same time, however, the group has also suggested that the authorisation process is simplified, with the development of a unique mechanism for authorisation (regardless of whether it concerns votes or elections), requiring for example that cantons are authorised only by the Federal Council, and clarifying the distinction between authorisation and certification. At the same time, they stress that external controls should remain, but that the control by the *groups d'acompagnement* should not be compulsory.

More important, the group endorses the decision by the Federal Council which requires the publication of the source code of those solutions providing complete verifiability. In June 2018 the Federal Council amended VEleS, which now requires that solutions providing complete verifiability publish their source code (Art. 7a and 7b). This obligation is set for these solutions regardless of whether the system is going to be used by the whole cantonal electorate or not. In addition, a public intrusion test is required, which will allow interested parties to try to hack the systems that want to be certified at the third level.

The group has also widely addressed the issue of dematerialisation. They have concluded that fully achieving this goal would affect the voting systems already authorised[10]. In turn, identification issues would arise and voters using online voting would not be able to cancel their online vote using another channel (e.g. by post). Therefore, they have recommended moving towards the so-called paper-saving electronic voting (voters' identification card including return codes is the only printed document which is delivered) and to conduct pilots to test the full dematerialisation of voting (no voting materials are printed at all). At the same time, vendors have already started looking for alternative voting protocols that are at least as robust as Return Codes (i.e. difficult collusion) but that do not require the use of paper at all.

[10] Since they are based on using Return Codes for providing individual verifiability which make use of paper-based voting cards [16].

5 Conclusions

Switzerland has been always one of the main references on the introduction of internet voting. The main approach used by the Swiss government has been to prioritise security instead of speed of adoption. For this reason, Switzerland started limiting the number of cantons and population who could use internet voting as a measure to minimise the impact that any misuse or attack could have had during its implementation. After carefully analysing the previous experiences, the Swiss government has been taking progressive steps on the deployment of online voting in the whole territory (e.g. opening the possibility for an increasing number of cantons to implement online voting). The last major change happened in 2014 with the definition of a new regulation that allows cantons to increase or even eliminate the electorate limits to introduce online voting. The main revolutionary aspects of this regulation are its security requirements and authorisation process. These allow to increase the number of voters who can make use of the electronic channel to cast their votes. In fact, it has been the first standardisation and certification framework for online voting systems defined to date. Other countries use the tender process to evaluate the security of voting systems (e.g. Norway) while other just define the mechanism that need to be implemented instead of evaluating different technologies (e.g. Estonia).

After evaluating how the Swiss government has conducted the introduction of online voting in the country, we can identify the following best practices:

- Start the introduction of online voting with specific collectives to reduce the initial risks inherent to the introduction of new election technologies and systems.
- Make gradual and incremental deployment steps after evaluating the results of experiences.
- Involve security and election management bodies in the evaluation process and the definition of online voting requirements.
- Set-up a clear framework for evaluating and certifying the security of the online voting systems, instead of designing closed solutions.
- Periodically review the legal and technical requirements for regulations to be flexible enough ahead of new improvements on technology, security, and auditability of voting systems.

References

1. Pratchett, Lawrence and Wingfeld, Melvin: Electronic Voting in the United Kingdom: Lessons and limitations from the UK experiences. In: Kersting, Norbert and Baldersheim, Harald (eds): Electronic Voting and Democracy: A Comparative Analysis. (2004).
2. Solop, Frederic I.: Electronic Voting in the United States: At the Leading Edge or Lagging Behing? In: Kersting, Norbert and Baldersheim, Harald (eds): Electronic Voting and Democracy: A Comparative Analysis. (2004).
3. OSCE/ODIHR: Swiss Confederation Federal Elections 23 October 2011. OSCE/ODIHR Needs Assessment Mission Report. (2011).

4. Braun, Nadja: E-Voting: Switzerland’s Project and their Legal Framework – in a European Context. In: Prosser, Alexander; Krimmer, Robert (eds): Electronic Voting in Europe: Technology, Law, Politics and Society. Bonn, pp. 43-52 (2004).
5. Driza Maurer, Ardita: Internet voting and federalism: the Swiss case. In: Barrat i Esteve, Jordi (coord): El voto electrónico y sus dimensiones jurídicas: entre la ingenua complacencia y el rechazo precipitado. Madrid, pp. 261-288 (2016).
6. Swiss Federal Council. Federal Act on Political Rights (FAPR) of 25 May 1978 (Status as of 11 November 2015)
7. Mendez, Fernando and Serdült, Uwe: From Initial Idea to Piecemeal Implementation: Switzerland's First Decade of Internet Voting Reviewed. In: Zissis, Dimitrios and Lekkas, Dimitrios (eds): Design, Development, and Use of Secure Electronic Voting Systems. (2014).
8. Krimmer, Robert: The Evolution of E-voting: Why Voting Technology is Used and How it Affects Democracy. (2002)
9. Luechinger, Simon; Rosingerand, Myra and Stutzer, Alois: The Impact of Postal Voting on Participation: Evidence for Switzerland. Swiss Political Science Review, Volume 13, Issue 2. (2011).
10. Geser, Hans: Electronic Voting in Switzerland. In: Kersting, Norbert and Baldersheim, Harald (eds): Electronic Voting and Democracy: A Comparative Analysis. (2004).
11. Germann, Micha and Serdült, Uwe: Internet voting and turnout: Evidence from Switzerland. In: Electoral Studies, 47, pp. 1-2. (2017).
12. Serdült, Uwe; Germann, Micha; Mendez, Fernando; Portenier, Alicia and Wellig Christoph: Fifteen Years of Internet Voting in Switzerland: History, Governance and Use. In: Teran, Luis and Meier, Andreas (eds): ICEDEG 2015: Second International Conference on eDemocracy & eGovernment. (2015).
13. Swiss Federal Council. Ordinance on Political Rights (OPR) of 24 May 1978. (Status as of 15 January 2015).
14. Swiss Federal Chancellery. Federal Chancellery Ordinance on Electronic Voting (VEleS) of 13 December 2013. (Status as of 1 July 2018).
15. Gharadaghy, Rojan and Volkamer Melanie (2010) Verifiability in Electronic Voting - Explanations for Non- Security Experts. In: Krimmer, Robert and Grimm Rüdiger (eds): Proceedings of the 4th Conference on Electronic Voting, LNI GI Series, Bregenz, Austria, July 21-24, pp 151 - 162
16. Puiggali, Jordi; Cucurull, Jordi; Guasch, Sandra and Krimmer, Robert: Verifiability Experiences in Government Online Voting Systems. In: Krimmer, Robert; Volkamer, Melanie; Braun Binder, Nadja; Kersting, Norbert; Pereira, Oliver and Schürmann, Carsten (eds): Electronic Voting: Second International Joint Conference, E-Vote-ID 2017, Bregenz, Austria, October 24-27, pp: 248-263. (2017).
17. Swiss Federal Chancellery. Catalogue des exigences à remplir pour recourir au vote électronique lors de votations populaires fédérales. Version of June 2014.
18. Swiss Federal Chancellery. Mandat du Groupe d'experts Vote électronique: passage à la mise en exploitation et dématérialisation du vote. Version of 25 August 2017.
19. Rapport final du groupe d'experts Vote électronique (GE VE). Version of April 2018.
20. BSI-CC-PP-0037. Common Criteria Protection Profile for Basic set of security requirements for Online Voting Products. Version 1.0 of 18 April 2008.
21. Council of Europe. Recommendation Rec(2004)11 of the Committee of Ministers to member states on legal, operational and technical standards for e-voting.
22. Council of Europe. Guidelines on transparency of e-enabled elections.
23. Council of Europe. Certification of e-voting systems: Guidelines for developing processes that confirm compliance with prescribed requirements and standards

24. Galindo, David; Guasch, Sandra and Puiggalí, Jordi: 2015 Neuchâtel's Cast-as-Intended Verification Mechanism. In: Haenni, Rolf; Koenig, Reto and Wikström, Douglas (eds): E-Voting and Identity. VoteID 2015. Lecture Notes in Computer Science, vol 9269. Springer, Cham, pp: 3-18. (2015).
25. Puiggalí-Allepuz, Jordi and Guasch-Castelló, Sandra: Privacy and Anonymity Management in Electronic Voting. In: UPGRADE. The European Journal for the Informatics Professional. Vol XI, issue No. 1, pp. 49-65. (2010).
26. OSCE/ODIHR: Swiss Confederation Federal Assembly Elections 18 October 2015. OSCE/ODIHR Election Expert Team Final Report. (2015).
27. Swiss Federal Chancellery. Essais de vote électronique. Essais 2018. Données essais 04.03.2018. (2018).
28. Scytl Secure Electronic Voting. Swiss Online Voting System. Cryptographic proof of Individual Verifiability. (2017). Available at: <https://www.post.ch/-/media/post/evoting/dokumente/swiss-post-cryptographic-proof-of-individual-verifiability.pdf?la=en&vs=1>
29. Cortier, Véronique; Turuani, Mathieu and Galindo, David: Analysis of Cast-as-Intended Verifiability and Ballot Privacy Properties for Scytl's Swiss On-line Voting Protocol using ProVerif (version 2). (2017). Available at: <https://www.post.ch/-/media/post/evoting/dokumente/analysis-verifiability-and-privacy-properties-for-swiss-post-voting.pdf?la=en&vs=1>
30. Republic and Canton of Geneva. Poursuite du développement de CHvote, le système de vote électronique genevois. (2016). Available at: <http://www.ge.ch/conseil_etat/2013-2018/ppresse/20160323.asp#P7>
31. Haenni, Rolf; Koenig, Reto and Dubuis. Eric: Cast-as-intended verification in electronic elections based on oblivious transfer. In: Barrat, Jordi; Krimmer, Robert; Volkamer, Melanie; Benaloh, Josh; Goodman, Nicole; Ryan, Peter; Spycher, Oliver; Teague, Vanessa and Wenda, Gregor (eds): E-Vote-ID 2016. 12th International Joint Conference on Electronic Voting. LNCS 10141, pages 277–296, Bregenz, Austria. (2016).
32. Republic and Canton of Geneva. E-voting system – CHVote. Protoype of the next generation protocol. Available at: <https://github.com/republique-et-canton-de-geneve/chvote-protocol-poc>
33. Brelle Achim and Truderung, Tomasz: Cast-as-Intended Mechanism with Return Codes Based on PETs. In: Krimmer, R., Volkamer M., Braun Binder N., Kersting N., Pereira O., Schürmann C. (eds) Electronic Voting. E-Vote-ID 2017. Lecture Notes in Computer Science, vol 10615. Springer, Cham (2017)
34. Brelle A., Truderung T. Cast-as-Intended Mechanism with Return Codes Based on PETs. In: Krimmer, Robert; Volkamer, Melanie; Braun Binder, Nadja; Kersting, Norbert; Pereira, Oliver and Schürmann, Carsten (eds): Electronic Voting: Second International Joint Conference, E-Vote-ID 2017, Bregenz, Austria, October 24-27. (2017).
35. Swiss Federal Chancellery. Consultation nouvel instrument de planification - rapport d'évaluation.

Email Voting in Indiana Elections

Jason Ryan Thompson[1,2,3][0000−0003−0730−8334]

[1] KU Leuven, Oude Markt 13 - bus 5005, 3000 Leuven, Belgium
[2] University of Muenster, Schlossplatz 2, 48149 Muenster, Germany
[3] Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia
jason.thompson@student.kuleuven.be

Abstract. The State of Indiana currently manages a vote by email system for overseas and military residents in all state and federal elections. This system, which requires voters to waive their right to privacy, introduces several problems which impede traditional pillars of modern democracies, namely ballot secrecy and electronic privacy. This paper explores the impetus for creating this system, the security implications for running a vote by email service, and the ramifications on voters that use such a system.

Keywords: ballot secrecy · data privacy · e-voting · elections · email voting · open voting · secret ballot.

1 Introduction

Well run elections are the cornerstone of a well functioning democracy. These elections should ideally be conducted in a way that treats all voters equally so that the whole public is fairly represented. In order for elections to be run in increasingly more efficient ways, technology has been introduced over the past century to make elections faster, more accurate, and more open. Tabulation machines introduced in the 1950s and 1960s made vote counting a quicker process while electronic voting machines introduced in the past decades increased the efficiency of voting exponentially. The next step in the technological evolution of elections is to make them as accessible as possible regardless of the location of the voter. In many jurisdictions, sending your ballot by mail has become a popular way for voters to more conveniently participate in the electoral process from home or abroad. Now, many governments are taking this method of remote voting a step further and have developed systems for voting over the internet, with varying success and drawbacks.

This paper will examine one such system: the vote by email system introduced in the State of Indiana in the United States. Indiana's email voting system is a three part process in which a PDF ballot is emailed to the voter, the voter marks up the ballot in physical or digital form, and finally it is emailed back to the local election board. The paper will question and explore the drivers for why such a system was introduced, the technological and security drawbacks, and whether the system in place can be considered technological progress or regression. The conclusions of this exploration will be useful when comparing the value and drawbacks of internet-based voting systems in other jurisdictions.

2 The Secret Ballot and Email Security

This section will examine the two main subjects that will be utilized in this paper's research: the concept of the secret ballot in elections and the security & privacy of email. First, the concept of the secret ballot will be explained followed by a discussion of the specific reasons why a voter would waive their right to a secret ballot. Then, there will be a discussion about voter's opinions about ballot secrecy. Finally, a discussion about the security of electronic communication methods like email will be presented.

2.1 Secret Ballot

The majority of scholars agree that utilizing the secret ballot in elections is one of the cornerstones of a healthy democracy [18]. The United Nations defined it as a core human right because "the will of the people shall be the basis of the authority of government" [2]. The freedom from influence that the secret ballot gives the voter allows them to freely choose their government representation without retribution from outside forces.

Brettschneider explains and justifies the use of the secret ballot in government elections:

> In deciding how to vote, citizens are entitled to freedom from coercion and to a "private space" in which to make up their own minds through the exercise of political judgment. The privacy of the voting booth serves to enhance this sense that we are free to make our own decisions without external coercion. This rationale also extends beyond procedural protections in the paradigmatic case of voting to the general role privacy rights play in a citizens capacity to think of themselves as rulers. [8]

The secret ballot is one of the primary tools used by government to legitimize themselves to their citizenry by making the choice of the people absolute, autonomous, and anonymous. Actors that would want to influence or intimidate the voter will not be able to physically witness the actual act of voting, giving the voter the ability to make an independent choice [18]. When contrasting the secret ballot with a more open system championed by Mill, Bentham posited that "there is a moral imperative to secure greatest utility (or happiness) for the greatest number" by honestly polling the internal wishes of the populous rather than openly encouraging a common normalized moral position [19]. In short, elections should be the collection of many individual views declared in secret instead of a public collective view. Small scale coerced votes in this system would be minimal and would not affect the overall result.

Waiving the Right to a Private Ballot In certain situations, the state may require a voter to waive their right to a secret ballot in order to accommodate them casting a vote in an alternative way. Saglie and Segaard cite an example from Birch and Watt in which a court in Ireland decided that "there was reason to depart from the principle of ballot secrecy in order to make it possible for certain groups to be able to vote at all [but] this is not a reason for departing from the principle in other contexts" [18]. In the modern day, a common reason for waiving this right is to be able to vote through, depending on the jurisdiction, a mail-in ballot or electronically through the internet or fax. The convenience factor of being able to receive and send a ballot electronically enables voters to more efficiently participate in the democratic process without the delay of the mailing system. Waiving their right to privacy is a necessary formality to achieve this speed.

Another common example for waiving the right to a secret ballot is to accommodate a disability that a voter may have that could prevent them from voting alone. A sight impaired voter would need a person to read and help fill out a ballot if a braille version was not available. A mobility impaired person may not be able to fill out the ballot at all. In addition, with the introduction of new and more complicated voting machines, those unable or unwilling to learn and utilize the technology would require assistance as well. If polling stations are not able to provide solutions to those that need alternative methods to vote, the right to vote in an election surpasses the right to a secret ballot.

Both of these examples are cases in which ballot secrecy is surrendered, but there is one key difference that makes the former a more fundamental problem for democracy. When a voter surrenders their right to privacy at the voting booth, they are aware of the person to whom they are surrendering. When surrendering by form to vote by email, they are surrendering their right to privacy to the state apparatus itself with no way to know who can view their ballot. This is in contrast to voting by mail where there is generally still some separation between voter and ballot, maintaining the secrecy of the ballot.[4]

Modern Indifference for the Secret Ballot While the academic and political arguments for the secret ballot have now been discussed, the opinions of actual, modern day voters have not. Gerber et al conducted a survey of American voters in 2008 which produced results that offer a new viewpoint on the value of privacy in the voting booth [14]. This first question, aiming to see if voters psychologically believe their votes are kept secret, was posed to the survey recipients:

> As far as you know, when you go to a polling place and vote, are your choices about which candidate you voted for kept secret unless you tell someone, or are your choices not kept secret?

[4] For example, in the State of Indiana, absentee ballots received by mail are sealed inside 'security envelopes.' The envelope is separated from the voter application form (containing identifying information for the voter) before it is opened and the vote is counted [17].

25.5% of responders assumed that their choices were “Not kept secret” [14]. A second question aimed to see if social factors also undermined the secret ballot:

> Either before or after an important election, do you mention which candidate you prefer or voted for to at least one other person, for instance a close friend or family member?

48.8% answered “Almost all the time” and 23.8% answered “Most of the time” [14]. The results of this study conclude that a large amount of active voters either do not believe in the integrity of the secret ballot or, due to social factors or pressure, reveal their choice voluntarily. Ultimately, it casts voting as more of a social act where voter’s choices are monitored & judged rather than a private & secret act, and it questions whether or not the formal secret ballot can function within the modern election environment.

2.2 Security of Electronic Communication

Vinton Cerf, widely considered to be one of the “fathers of the internet”, wrote when discussing email privacy: “Is email ever private?” [9]. He notes that email privacy is dependent on many uncontrollable factors, including the security of both the sender and the recipient, the path taken to the recipient, and the use of security measures like cryptography by all actors involved. Even though email has been in wide use for over two decades, a set of security standards to protect the message and its contents has not been established, leaving one of the most used communication protocols as one of the least secure.

There are two main security flaws that make email an inappropriate platform for sensitive communication: the possibility that messages can be “intercepted or altered” and the relatively simple act of modifying sender information so messages appear to be sent from someone else [6]. Due to the nature of how email traverses the internet, a message is copied and forwarded through many intermediate servers between the sender’s computer and the recipient’s computer, usually entirely in plaintext. While these intermediate servers are supposed to delete messages after they are forwarded along, a server operated by a malicious entity can be configured to archive these messages. A much more common and low-cost alternative for hackers to intercept email is to simply use packet sniffing software on a normal laptop in a public environment to collect email data transmitted through wireless access points. Regardless of the collection method, these archived messages can be altered and forwarded in place of the original message.

It is also trivial for a malicious actor to alter the message’s metadata to claim a message comes from another person [15]. These “spoofed” messages can then be sent to unsuspecting recipients posing as another entity and request them to send sensitive data back to the hacker. Kruck and Kruck explain that “email spoofing is possible because Simple Mail Transfer Protocol (SMTP), the main protocol used in sending email, does not include an authentication mechanism” [15]. The blind trust that many users place in email is similar to the technical underpinnings of how the protocol operates, and because of this, email is the prime platform for those with malicious intentions to operate.

2.3 Conclusion

This section discussed two broad topics that are important to the research in this paper: the concept of the secret ballot and the security of electronic communication, specifically email. A background in these concepts is necessary to analyze the main topic of the paper, the vote by email system in the State of Indiana.

The widespread use of email, along with its inherent security issues, by actors in both the public and private sector can be linked with the convenience factor and ubiquity of the platform. It is an obvious choice for a government to communicate with its citizens due to these factors. In the case of the Republic of Estonia, the government even provides each resident a government-issued email address in which this communication is conducted [5]. However, any sensitive data should not be sent using the email platform due to the lack of a comprehensive common security layer. Therefore, the use of email as the medium for government to citizen communication is, as with many technologies, a struggle between security and convenience.

3 Voting by Email in Indiana Elections

This section begins the discussion of the main topic of this paper: the use of email as a method of submitting a vote in Indiana elections. First, a brief discussion of the federally mandated push towards online voting methods will occur followed by Indiana's implementation of the federal mandate in its law. Then, the security implications of an email based voting system will be discussed. This section ends with a brief and more broad discussion about the problems with providing private and confidential government services over the internet in the United States.

3.1 Federal Legislation Regarding Online Voting

Online voting in the United States is an initiative started primarily by the 2010 Military and Overseas Voter Empowerment Act (MOVE) [10]. As the title of the act illustrates, online methods of voting are only meant for United States citizens living outside its borders, not those within the country. Coleman summarizes the act's main provisions regarding online voting:

> – States are required to establish procedures to permit absent uniformed services voters and overseas voters to request voter registration and absentee ballot applications by mail and electronically for all federal elections.
> – States are required to establish procedures to transmit, by mail and electronically, blank absentee ballots to absent uniformed services voters and overseas voters for federal elections. [10]

These provisions from the MOVE Act are concerned with only half of the voting process: getting the ballot to the overseas voter. The process of returning the filled out ballot back to the relevant election board remained ambiguous and up for interpretation on a state-by-state basis. This led to a fracturing of online voting services offered by states with some accepting returned email ballots while others requiring it be printed and physically mailed back to the United States.

3.2 Law Regarding Online Voting in Indiana

The Indiana Constitution explicitly states that all elections within the state are to be done by ballot, not *viva voce* [1]. The remainder of legislation regarding overseas citizen's election regulation and rights is codified in Title III of the Indiana Code. First, Indiana's implementation of the MOVE Act will be examined followed by an overview of the "Absentee Voter's Bill of Rights." Then, the procedure for voting by email will be discussed.

Indiana's implementation of the federal MOVE Act states that "to implement 52 U.S.C 20302, electronic mail, fax, and web publication are designated as means of communication for a voter to request a voter registration application and an absentee ballot application" [4]. Indiana's method for voters to return their completed ballot is the following:

> The county election board shall by fax or electronic mail transmit an absentee ballot to and receive an absentee ballot from an absent uniformed services voter or an overseas voter by electronic mail or fax at the request of the voter indicated in the application filed under this section. If the voter wants to submit absentee ballots by fax or electronic mail, the voter must separately sign and date a statement submitted with the electronic mail or the fax transmission that states substantively the following: "I understand that by faxing or e-mailing my voted ballot I am voluntarily waiving my right to a secret ballot." [4]

The Absentee Voter's Bill of Rights Title III states that a county election board must "prescribe a statement known as the Absentee Voter's Bill of Rights" that will be included with each absentee ballot sent to a voter [4]. This document must contain "the rights and responsibilities of the voter when casting and returning the absentee ballot" along with "a summary of Indiana and federal laws concerning providing assistance to the voter, completion of the ballot in secret, intimidation of voters, and the return of the absentee ballot to the county election board" [4]. The actual document, in use and not modified since 2009, is a bulleted list of statements that affirm the voter's right to secrecy, right from coercion or electioneering, and to remind the voter that it is a crime "for a person to accept any payment or property for casting an absentee ballot" [3]. When detailing instructions for returning a marked ballot, it says the following:

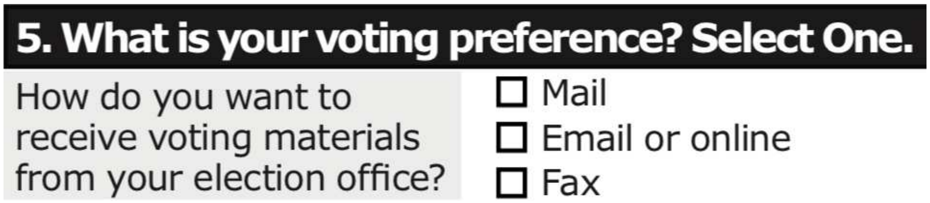
> After you have sealed your absentee ballot inside the envelope you return the envelope to the county election board in the following manner:

> 1) by U.S. mail or a bonded courier company; 2), by hand-delivering the envelope yourself; 3) by delivering the envelope to a member of your household (or to a person to whom you have given your power of attorney). If you deliver the sealed envelope to a household member or your power of attorney, that person must deliver your ballot to the county election board in person, by U.S. mail, or by using a bonded courier company. [3]

The ability to receive or submit a vote through email or fax is not mentioned in the Absentee Voter's Bill of Rights along with the obligation that one must waive their right to a secret ballot if a vote is submitted online.

Process to Vote by Email This section will describe the process used to vote by email in an election run by the State of Indiana. To request a ballot to be delivered by email, a voter must first fill out and email the Federal Post Card Application (FPCA) form to their county voting board [17]. Figure 1 shows the question on the FPCA form where the voter is given the option to receive the ballot by email. Once the relevant election board receives the request for an email ballot, they send via email the following documents:

- The ballot for the election in PDF format (see Figure 2)[5]
- Form ABS-9: Cover Sheet and Affidavit for Absent Uniformed Services and Overseas Voter (see Figure 3)
- Additional instructions from the county board (but not the aforementioned Absentee Voter's Bill of Rights)



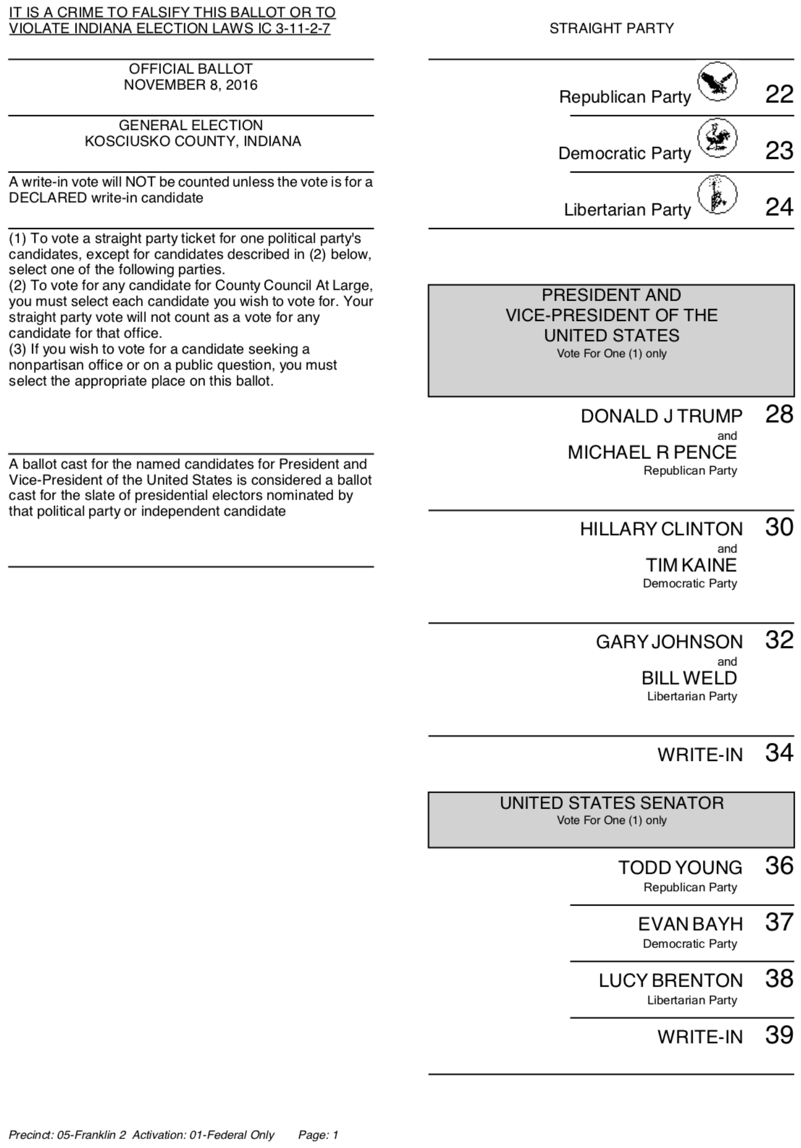
5. What is your voting preference? Select One.

How do you want to receive voting materials from your election office?	□ Mail □ Email or online □ Fax

Fig. 1. Question 5 of the FPCA form showing the choice to receive a ballot through email

The voter then marks the ballot (digitally or on a printed version), fills out the ABS-9 form, and then sends digital copies of both back to the county election

[5] Generally, the ballot the voter receives will be a "regular ballot" containing races at local, state, and federal level. If the voter checks the box for "I am a U.S. citizen living outside the country, and my return is uncertain.", only a federal ballot will be sent. If the voter checks "I am a U.S. citizen living outside the country, and I have never lived in the United States.", no ballot is sent and the application is rejected [17].



IT IS A CRIME TO FALSIFY THIS BALLOT OR TO
VIOLATE INDIANA ELECTION LAWS IC 3-11-2-7

OFFICIAL BALLOT
NOVEMBER 8, 2016

GENERAL ELECTION
KOSCIUSKO COUNTY, INDIANA

A write-in vote will NOT be counted unless the vote is for a DECLARED write-in candidate

(1) To vote a straight party ticket for one political party's candidates, except for candidates described in (2) below, select one of the following parties.
(2) To vote for any candidate for County Council At Large, you must select each candidate you wish to vote for. Your straight party vote will not count as a vote for any candidate for that office.
(3) If you wish to vote for a candidate seeking a nonpartisan office or on a public question, you must select the appropriate place on this ballot.

A ballot cast for the named candidates for President and Vice-President of the United States is considered a ballot cast for the slate of presidential electors nominated by that political party or independent candidate

STRAIGHT PARTY

Republican Party 22

Democratic Party 23

Libertarian Party 24

PRESIDENT AND
VICE-PRESIDENT OF THE
UNITED STATES
Vote For One (1) only

DONALD J TRUMP 28
and
MICHAEL R PENCE
Republican Party

HILLARY CLINTON 30
and
TIM KAINE
Democratic Party

GARY JOHNSON 32
and
BILL WELD
Libertarian Party

WRITE-IN 34

UNITED STATES SENATOR
Vote For One (1) only

TODD YOUNG 36
Republican Party

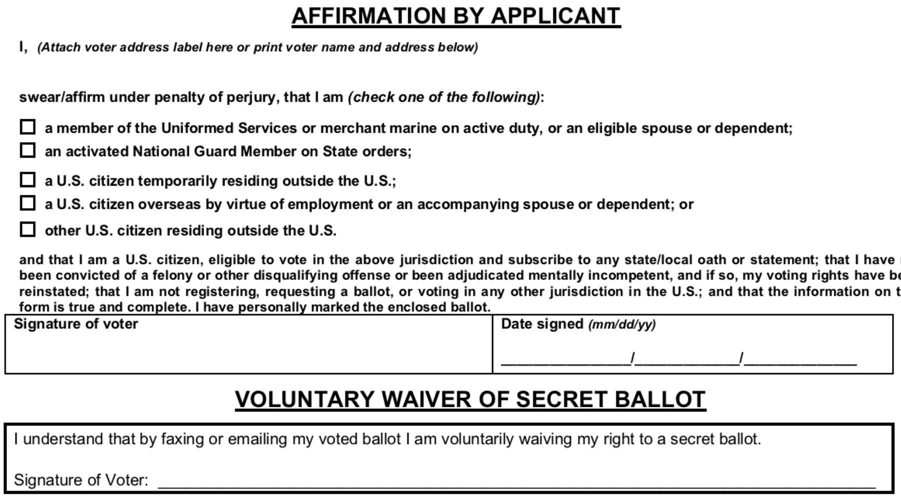
EVAN BAYH 37
Democratic Party

LUCY BRENTON 38
Libertarian Party

WRITE-IN 39

Precinct: 05-Franklin 2 Activation: 01-Federal Only Page: 1

Fig. 2. Page 1 of the 2016 General Election Federal ballot sent by email to voters in the Franklin 2 district of Kosciusko County, Indiana



AFFIRMATION BY APPLICANT

I, ***(Attach voter address label here or print voter name and address below)***

swear/affirm under penalty of perjury, that I am *(check one of the following)*:

- □ **a member of the Uniformed Services or merchant marine on active duty, or an eligible spouse or dependent;**
- □ **an activated National Guard Member on State orders;**
- □ **a U.S. citizen temporarily residing outside the U.S.;**
- □ **a U.S. citizen overseas by virtue of employment or an accompanying spouse or dependent; or**
- □ **other U.S. citizen residing outside the U.S.**

and that I am a U.S. citizen, eligible to vote in the above jurisdiction and subscribe to any state/local oath or statement; that I have not been convicted of a felony or other disqualifying offense or been adjudicated mentally incompetent, and if so, my voting rights have been reinstated; that I am not registering, requesting a ballot, or voting in any other jurisdiction in the U.S.; and that the information on this form is true and complete. I have personally marked the enclosed ballot.

Signature of voter	**Date signed** ***(mm/dd/yy)*** ______________/___________/____________

VOLUNTARY WAIVER OF SECRET BALLOT

I understand that by faxing or emailing my voted ballot I am voluntarily waiving my right to a secret ballot.

Signature of Voter: ____

Fig. 3. The bottom section of the ABS-9 form showing the "Voluntary Waiver of the Secret Ballot"

board by email. A bipartisan team then transfers the vote marks seen on the digital ballot onto a physical absentee voter ballot and is initialed by both parties [17]. This new ballot along with the original emailed ballot are placed in a regular absentee vote envelope marked as "Absentee Ballot Received By Email Or Fax". From this point on, the vote is treated like a regular absentee ballot.

4 Analysis

4.1 Security Implications for Email Voting

As detailed in the previous discussion of email security, anything sent by email that is not cryptographically encoded with a private key cannot be considered secure. This is due to the architecture of the email protocol itself and it not being designed from the start to secure the integrity of messages. This must be considered a permanent limitation of the system as it cannot easily be addressed without a significant overhaul of the infrastructure of the SMTP protocol itself.

Building a government service that requires the secrecy and integrity of messages on the email system was destined to failure from the outset. Email messages sent with PDF ballots intended to filled out by the voter can theoretically be intercepted and altered as well as emails with filled out ballots sent back to the election board. While the former is likely to be caught by the election board, the latter is much more difficult to detect. Given the recent climate of claims of election fraud and vote rigging in the United States, such a flawed and easily exploitable voting system further damages the integrity of elections in the eyes of both politicians and voters alike [11]. Given that similar email voting systems exist in 22 other states, this election security issue is likely to continue and increase in controversy.

While the security of email itself is known to be flawed, a second, arguably more important breach of security is also occurring with the use of such a flawed system: the privacy of the secret ballot. While it is true that those in Indiana who vote by online means must waive this right, the reality of a subset of voters in a secret ballot system being exposed can lead to repercussions for the electoral choices of these voters. Indiana law is ambiguous whether the waiving of the voter's right to secrecy also waives the penalties in place for deliberately exposing the secret ballot of a traditional voter. Therefore, it is possible that there is no legal protection for those persecuted for their vote. This lack of protection and privacy for voters is a significant right to waive for the convenience of voting online.

4.2 Improving Electronic Voting in the United States

The current system for electronic voting by email has several flaws which can be improved upon. One obvious improvement is to move all sensitive communication completely off the email platform. In five states (Alabama, Alaska, Arizona, Missouri, and North Dakota), an internet portal has been launched to distribute and accept votes [13]. While this is a better system when compared to email, there are still significant flaws in a portal setup, the biggest of which is the lack of a secure digital identification system for United States citizens [16]. The de-facto US national ID number, the social security number, is used by current online government services as an identifier, but it is a system that was never designed with security in mind. In fact, the remaining integrity of the already weak system was completely destroyed when almost 150 million identity records (including social security numbers) were stolen from the private credit agency Equifax in 2017 [12]. Therefore, since voters cannot securely identify themselves online and considering the current climate of maintaining higher security regarding possible election fraud, electronic voting systems using insecure technology including email, fax, or an online portal, should be discontinued until a secure digital identification system can be developed. Once such a system is built, this can open the door for many possibilities for online government services, including voting.

4.3 Addressing Research Questions

This section will directly answer the research questions posed at the beginning of this paper using the research on online voting in Indiana contained in the previous sections.

What were the drivers for creating an email-based voting system for Indiana elections? The email-based voting system developed in Indiana (along with similar systems in other states) began as a result of the adoption of the

Military and Overseas Voter Empowerment Act at the federal level. This act mandated that all states develop a method of registering voters and delivering ballots electronically to citizens living outside of the United States. It did not mandate a specific method for returning these ballots. Some states required these electronic ballots to be printed and returned via the postal system. In Indiana and several other states, an option for voters to email or fax their ballots was also offered. The caveat to this more convenient method of submitting the ballot was that the voter must waive their right to a secret ballot.

What are the security implications for people voting through email in Indiana elections? Email was, for its time, an efficient and convenient way to deliver messages between users over the internet. This convenience, however, allows the email protocol to be easily exploited by malicious actors. These flaws in the core of the email system makes it a poor choice for messages that need to maintain their integrity and secrecy. Unencrypted email messages can easily be intercepted and altered which is simply not an acceptable risk for maintaining the democratic process. In addition, since ballots sent by email are in plaintext and the contents of the voter's choice are able to be seen by anyone, the benefits and protections of the secret ballot are no longer in place. This could make the voter an object of persecution if their private choice is made known publicly. The only benefit of an email based system for vote submission is the convenience of casting a vote to the election authorities near instantaneously. This convenience brings with it numerous technical and social problems which make it poor solution for electronic voting.

Can Indiana's vote by email system be considered a failed attempt at an e-government service? Assar, Boughzala, and Boydens write that a successful e-government service should have the following characteristics: [7]

1. The usage of Internet technologies to deliver services online and to rationalize, redesign, and significantly improve public administrative processes;
2. The reorganization of public institutions in order to reduce cost and to enhance the quality and efficiency of services offered to citizens, companies, and other governmental partners;
3. The development of new democratic spaces in which relations among public institutions, citizens, and enterprises are redefined according to a participatory perspective.

The email voting system in Indiana fails to meet aspects of all three of these characteristics. First, while vote by email utilizes the internet to deliver a formally analog service online, it does so in a way that does not "significantly improve" the voting process. In fact, due to the security implications for using email as a medium for transmitting sensitive information, it significantly worsens the voting process from the standpoints of election integrity and voter privacy.

Regarding the second characteristic, the email voting system did not require a reorganization of election boards, and the only cost savings to speak of were that of the postage on mailed ballots. While an argument can be made for improved efficiency using email voting, it does not improve the quality of the service as previously explained.

Finally, and most importantly, the email voting system has the potential to close democratic spaces between citizens and the public sector due to the forced waiver of ballot secrecy. The system and its users are a small subset of the full Indiana electorate now, but if rolled out to a larger audience, the implications of a chilling effect on democracy are quite probable. Since a system like this is an obvious target for hackers, the likelihood of a major data breach is high.

5 Conclusion

Elections should be as accessible as possible to all citizens of a jurisdiction regardless of location. Technology can be used to break down barriers of distance, but only if it used carefully and with full thought of the implications of its use. All votes, regardless of method or use of technology, have the potential to be mishandled, miscounted, or altered, and electronic voting technology can be used to mitigate some of these longstanding issues. However, not all technology is created equally, and some voting innovations may bring more problems than they solve.

Indiana's vote by email system does allow more voters to more efficiently participate in elections. It removes the delay to deliver and submit a physical ballot and allows all citizens, regardless of where they are in the world, to exercise their right to vote. It also introduces problems of compromised election integrity and voter secrecy as a aftereffect of this gain in efficiency. Solving these technological problems is an issue much larger than the State of Indiana itself. The lack of a secure federal digital identification system for United States citizens hinders e-government progress not only in voting but also in all government services. With the 2017 Equifax breach, more attention was given to replacing the long overburdened social security number identifier with a more secure and reliable from of ID. If the United States government follows through and creates such a system, voting by email may become a distant footnote in American election history.

References

1. Indiana constitution (1851)
2. Universal declaration of human rights (1948)
3. The absentee voter's bill of rights (2009), https://www.in.gov/sos/elections/files/Absentee_Voter_Bill_of_Rights_(2009_revision).pdf
4. Indiana code (2017)
5. Using eesti.ee address (2017), https://www.eesti.ee/eng/topics/riigiportaali_abi/ametliku_e_posti_seadistamine
6. Anderson, J.C.: Transmitting legal documents over the internet: How to protect your client and yourself. Rutgers Computer & Tech **27**(1), 1–50 (2001)

7. Assar, S., Boughzala, I., Boydens, I. (eds.): Practical Studies in E-Government: Best Practices from Around the World. Springer (2011)
8. Brettschneider, C.L.: Democratic Rights: The Substance of Self-Government. Princeton University Press, Princeton (2007)
9. Cerf, V.G.: When email isn't private. Communications of the ACM **59**(12), 15 (2016)
10. Coleman, K.J.: The uniformed and overseas citizens absentee voting act: Overview and issues. Congressional Research Service pp. 1–23 (2015)
11. Cottrell, D., Herron, M.C., Westwood, S.J.: An exploration of donald trump's allegations of massive voter fraud in the 2016 general election. Electoral Studies (In press) pp. 1–20 (2017)
12. Cowley, S.: 2.5 million more people potentially exposed in equifax breach (Oct 2 2017), https://www.nytimes.com/2017/10/02/business/equifax-breach.html
13. Fitzgerald, C., Smith, P., Goodman, S.: The secret ballot at risk: Recommendations for protecting democracy. Electronic Privacy Information Center pp. 1–141 (2016)
14. Gerber, A.S., Huber, G.A., Doherty, D., Dowling, C.M.: Is there a secret ballot? ballot secrecy perceptions and their implications for voting behaviour. British Journal of Political Science **43**, 77–102 (January 2013)
15. Kruck, G.P., Kruck, S.: Spoofing - a look at an evolving threat. The Journal of Computer Information Systems **47**(1), 95–100 (2006)
16. Ni, A.Y., Ho, A.T.K.: A quiet revolution or a flashy blip? the real id act and u.s. national identification system reform. Public Administration Review **68**(6), 1063–1078 (2008)
17. Nussmeyer, A., Simmons, D.: Nuts & bolts of abs administration (2018), https://www.in.gov/sos/elections/files/Absentee.FINAL.pptx
18. Saglie, J., Segaard, S.B.: Internet voting and the secret ballot in norway: principles and popular understandings. Journal of Elections, Public Opinion and Parties **26**(2), 155–169 (2016)
19. Theuns, T.: Jeremy bentham, john stuart mill and the secret ballot: Insights from nineteenth century democratic theory. Australian Journal of Politics and History **63**(4), 493–507 (2017)

Protocols

Model Checking the SELENE E-Voting Protocol in Multi-Agent Logics

Wojciech Jamroga, Michal Knapik, and Damian Kurpiewski

Institute of Computer Science, Polish Academy of Sciences
{w.jamroga,michal.knapik,damian.kurpiewski}@ipipan.waw.pl

Abstract Selene is a recently proposed voting protocol that provides reasonable protection against coercion. In this paper, we make the first step towards a formalization of selected features of the protocol by means of formulae and models of *multi-agent logics*. We start with a very abstract view of the protocol as a public composition of a secret bijection from tracking numbers to voters and a secret mapping from voters to their choices. Then, we refine the view using multi-agent models of strategic interaction. The models define the space of strategies for the voters, the election authority, and the potential coercer. We express selected properties of the protocol using the strategic logic $\mathbf{ATL}_{\mathrm{ir}}$, and conduct preliminary verification by model checking. While $\mathbf{ATL}_{\mathrm{ir}}$ allows for intuitive specification of requirements like coercion-resistance, model checking of $\mathbf{ATL}_{\mathrm{ir}}$ is notoriously hard. We show that some of the complexity can be avoided by using a recent approach of *approximate model checking*, based on fixpoint approximations.

1 Introduction

Designing protocols for secure and verifiable voting is a difficult task. In this work, we present an attempt to use the techniques from multi-agent systems (MAS) in modeling and verification of e-voting. Agents in such systems are equipped with a larger degree of freedom than typical entities in a security protocol. They can have clearly defined objectives, capabilities, and knowledge about the world; they can also form coalitions working towards a joint goal. The benefits of MAS become especially noticeable in analysis of scenarios that involve interaction between human and technical agents, such as electronic voting.

Here, we come up with a simple MAS model of the recently proposed Selene protocol [19], and characterize several variants of coercion resistance with formulae of Alternating-time Temporal Logic (**ATL** [1]). Coercion resistance is essential in modern elections, and relies on the ability of voters to vote as they intend, and avoid the consequences of not obeying the coercer. Such requirements can be conveniently represented by **ATL** formulae following the scheme:

$$\neg\langle\langle \mathit{Coercer}\rangle\rangle \mathrm{F}\,\big(\textsf{election ends} \land (\textsf{voters have not obeyed}) \rightarrow (\mathit{Coercer}\ \textsf{knows})\big),$$

interpreted as *"The Coercer has no strategy to make sure that, when the election is over, he will detect disobedience of the coerced voters."* We use the semantics of **ATL**, based on memoryless imperfect information strategies, where agents' strategies assign choices of action to the agents' states of knowledge, rather than global states of the system. This variant of the logic is often referred to as $\mathbf{ATL}_{\mathrm{ir}}$ [20]. It is well known that model checking of $\mathbf{ATL}_{\mathrm{ir}}$ is $\mathbf{\Delta_2^P}$-complete [9,20] and does not have a natural fixpoint characterization [5]. To overcome the prohibitive complexity, we utilize a recently proposed idea of approximate verification based on fixpoint approximations of $\mathbf{ATL}_{\mathrm{ir}}$ formulae [10].

The article is organized as follows. We describe Selene, and discuss some of its formal aspects in Section 2. Then, we propose a multi-agent model of the protocol in Section 3. In Section 4, we present a brief introduction to $\mathbf{ATL}_{\mathrm{ir}}$, and propose some $\mathbf{ATL}_{\mathrm{ir}}$ formalizations of coercion-resistance. Section 5 reports our attempt at model checking the formulae from Section 4 in the models from Section 3. We conclude in Section 6, and discuss plans for future work.

1.1 Related Work

Over the years, the properties of *receipt-freeness* and *coercion resistance* were recognized as important for an election to work properly. They were studied and formalized in [3,7,8,13,18], see also [17,21] for an overview.

A number of papers used variants of epistemic logic to characterize coercion resistance [11,12]. Moreover, the agent logic **CTLK** together with the modeling methodology of interpreted systems was used to specify and verify properties of cryptographic protocols, including authentication protocols [4,16], and key-establishment protocols [4]. In particular, [4] used variants of the MCMAS model checker to obtain and verify models, automatically synthesized from high-level protocol description languages such as CAPSL, thus creating a bridge between multi-agent and process-based methods.

Our approach is closest to [21] where **ATL**-style formulae were used to encode different flavors of coercion resistance. However, the encodings in [21] were rather informal and imprecise, since neither formal semantics nor concrete model was given to interpret the formulae. In contrast, we use a precise semantics and provide a scalable class of models. Moreover, we use the formulae, the models, and the semantics to conduct verification of the protocol by model checking. Finally, [2] proposed a very simple attempt at model checking of Rivest's ThreeBallot protocol using $\mathbf{ATL}_{\mathrm{ir}}$, but the focus was on devising a model equivalence, and ThreeBallot served only to illustrate the idea.

2 Modeling Selene

We begin with a description of Selene, followed by a very abstract view of the conceptual backbone of the protocol. After that, we will move on to a more concrete model in Section 3.

2.1 Outline of Selene

Selene [19] has been proposed recently as a protocol for electronic voting targeted at low-coercion environments. The implemented cryptographic mechanisms should allow the voter to convince the coercer that the voter voted according to the coercer's request. One of the main advantages of the protocol is that, from the voter's perspective, the cryptography is put under the bonnet.

Roughly speaking, Selene works as follows. The Election Authority executes the initial setup of the system, which includes generation of the election keys and preparation of the cryptographic vote trackers, one for each voter. The trackers are then encrypted and mixed, and published on the Web Bulletin Board (WBB). The aim is to break any link between the voter and her encrypted tracker. Hence, the pool of trackers is public, while the assignment of the trackers is secret.

In the voting phase, each voter fills in, encrypts, and signs her vote. The signed and encrypted ballot is then collected by the system. After several intermediate steps, a pair (Vote_v, tr_v) is published in WBB for each $v \in \mathit{Voters}$, where Vote_v and tr_v are, respectively, the decrypted ballot and the tracker of v. At this stage, no voters know their tracker numbers. All the cast votes are presented in plaintext in WBB.

The final stage consists of the notification of tracker numbers. If the voter is not coerced, then she requests the special α_v term, which allows for obtaining the correct tracker tr_v. If some pressure was exerted on the voter to fill her ballot in a certain way, she sends a description of the requested vote to the election server. A fake α'_v term is sent, which can be presented to the coercer. The α'_v token, together with the public commitment of the voter, reveals a tracker pointing out to a vote compatible with the coercer's demand, assuming that there is one.

2.2 An Abstract View of the Protocol

We now propose a convenient way of describing the scheme behind Selene at the abstract level. Social choice can be seen as a function that, given a set of voters, produces a collective decision for the society. This can be decomposed into a mapping between voters and their individual choices, and a mapping from the choices to the collective decision. End-to-end voter-verifiable protocols strive to make the former individually verifiable (so that each voter can verify her part of the function), and the latter universally verifiable (so that the whole function can be verified by everybody). On the other hand, coercion resistant protocols strive to make the first part secret to anybody except for the voter in question. Selene's idea of how to combine the two objectives is to further decompose the connection between voters and their cast ballots by means of the trackers.

Formally, let *Voters*, *Trackers*, and *Choice* be three finite sets such that $|\mathit{Voters}| = |\mathit{Trackers}|$. The first part of the protocol corresponds to a random choice of a secret *tracker bijection* $\mathcal{F}_T : \mathit{Voters} \to \mathit{Trackers}$ that assigns a unique tracker to each voter. We denote the set of all such bijections by $\mathcal{T}$. Moreover, the final part of Selene can be presented as a *public bulletin function* $\mathcal{F}_P$: $\mathit{Trackers} \to \mathit{Choice}$ that assigns to each $tr \in \mathit{Trackers}$ the vote $\mathcal{F}_P(tr)$ cast by the owner of the tracker. The secret *choice function* $\mathcal{F}_I = \mathcal{F}_P \circ \mathcal{F}_T \colon \mathit{Voters} \to \mathit{Choice}$ connects voters with their ballots. The Election Authority is the only entity in the process that can observe the choice function. Note that this view can be applied to any voting system based on publicly visible trackers.

2.3 Combinatorial Aspects

Let $\mathcal{F}_P$, $\mathcal{F}_I$, and $\mathcal{F}_T$ be the public bulletin function, the choice function, and the tracker bijection. We will now estimate the range of uncertainty of the coercer, with the intuition that the less he knows about the real tracker assignment $\mathcal{F}_T$, the more room is available for coercion resistance. For each $tr, tr' \in \mathit{Trackers}$, let $tr \approx_{\mathcal{F}_P} tr'$ iff $\mathcal{F}_P(tr) = \mathcal{F}_P(tr')$, i.e., two trackers are "vote-equivalent" if they point to identical votes. Moreover, let $\mathit{Trackers}/\approx_{\mathcal{F}_P}$ denote the set of equivalence classes of $\approx_{\mathcal{F}_P}$. The uncertainty of the coercer can be measured by the number of permutations in the set of trackers, that he cannot distinguish from the actual tracker assignment.

Formally, let $\Pi(\mathit{Trackers})$ denote the set of all permutations of trackers, i.e., bijections $\pi \colon \mathit{Trackers} \to \mathit{Trackers}$. Now, $\mathcal{T}_{\mathcal{F}_P} = \{\pi \in \Pi(\mathit{Trackers}) \mid \mathcal{F}_P = \mathcal{F}_P \circ \pi\}$ is the set of all permutations of trackers that are consistent with the public outcome of the election. To see this, observe that for each $\pi \in \mathcal{T}_{\mathcal{F}_P}$ we have $\mathcal{F}_I = \mathcal{F}_P \circ \mathcal{F}_T = \mathcal{F}_P \circ \pi \circ \mathcal{F}_T$. The size of $\mathcal{T}_{\mathcal{F}_P}$ reflects the space of defensive capabilities against coercion, should a part of the secret tracker bijection become public, under the assumption that voting is one-shot rather than repeated.

Definition 1 (Anti-coercion space). *The* anti-coercion space *of an election, given the public bulletin function $\mathcal{F}_P$, is defined as $\mathit{acspace}(\mathcal{F}_P) = \{\mathcal{F}_P \circ \pi \mid \pi \in \Pi(\mathit{Trackers})\}$. Intuitively, $\mathit{acspace}(\mathcal{F}_P)$ corresponds to all the possible choice functions $\mathcal{F}_I$ consistent with $\mathcal{F}_P$.*

Theorem 1. *If the result of the election consists of n votes for candidates $\mathfrak{c}_1, \ldots, \mathfrak{c}_k$, s.t. each candidate $\mathfrak{c}_i$ got m_i votes, then $|\mathit{acspace}(\mathcal{F}_P)| = \frac{n!}{(m_1!)\cdot\ldots\cdot(m_k!)}$.*

Proof. Notice that $|\mathcal{T}_{\mathcal{F}_P}| = \prod_{\rho \in \mathit{Trackers}/\approx_{\mathcal{F}_P}} (|\rho|!)$. By the orbit-stabilizer theorem [14] we have $|\mathit{acspace}(\mathcal{F}_P)| = \frac{|\mathcal{T}|}{|\mathcal{T}_{\mathcal{F}_P}|}$, which concludes the proof.

Note that the space is typically vast, unless for very small elections or when almost all the voters voted for the same candidate. This is good news, as it makes it potentially hard for the coercer to obtain useful information about the real choices of the voters. On the other hand, a faithful representation of the coercer's state of knowledge leads to state-space explosion, which makes verification more complex. We will see it clearly in the next section.

3 Multi-Agent Model of Selene

In this section we present in detail a multi-agent model of Selene. We start with defining the formal structures used for modeling the entities participating in the protocol and their interactions, and move on to presenting the model of Selene, together with selected details of its implementation.

3.1 Models of Multi-Agent Interaction

Multi-agent systems are often modeled by a variant of transition systems where transitions are labeled with combinations of actions, one per agent. Moreover, epistemic relations are used to indicate states that look the same to a given agent. Formally, an *imperfect information concurrent game structure* or *iCGS* [1] is given by $M = \langle \mathbb{A}\mathrm{gt}, St, PV, V, Act, d, o, \{\sim_a \mid a \in \mathbb{A}\mathrm{gt}\}\rangle$ which includes a nonempty finite set of all agents $\mathbb{A}\mathrm{gt} = \{1, \ldots, k\}$, a nonempty set of states St, a set of atomic propositions PV and their valuation $V : PV \to 2^{St}$, and a nonempty finite set of (atomic) actions Act. The protocol function $d : \mathbb{A}\mathrm{gt} \times St \to 2^{Act}$ defines nonempty sets of actions available to agents at each state; we will write $d_a(q)$ instead of $d(a, q)$, and define $d_A(q) = \prod_{a\in A} d_a(q)$ for each $A \subseteq \mathbb{A}\mathrm{gt}, q \in St$. Furthermore, o is a (deterministic) transition function that assigns the outcome state $q' = o(q, \alpha_1, \ldots, \alpha_k)$ to each state q and tuple of actions $\langle\alpha_1, \ldots, \alpha_k\rangle$ such that $\alpha_i \in d(i, q)$ for $i = 1, \ldots, k$.

Every $\sim_a \subseteq St \times St$ is an epistemic equivalence relation with the intended meaning that, whenever $q \sim_a q'$, the states q and q' are indistinguishable to agent a. The *iCGS* is assumed to be *uniform*, i.e., $q \sim_a q'$ implies $d_a(q) = d_a(q')$.

It should be mentioned that *iCGS* generalize transition networks as well as normal form games, repeated games, and extensive form games. Moreover, it is possible to define the notions of *strategic play* and *strategic ability* in *iCGS*.

3.2 A Multi-Agent Model of Selene

In what follows, we describe our multi-agent model of Selene. The system consists of the set *Voters* of voter agents, the single *Coercer*, the Election Defense System *ElectionDS*, and the *Environment* agent. We denote the set of all these agents by *Agents*. The local states of each agent are defined by its local variables. A global state of the system is a valuation of local variables of all the agents. Each agent can observe its local variables and selected local variables of the *Environment*. For simplicity, we assume a single coercer. This precludes the case when, e.g., two coercers request two different votes from the same voter and then compare the results. We plan to study this type of interactions in the future.

The model is parameterized by the following natural numbers: n voters; k possible choices (i.e., the ways that a ballot can be filled); *maxCoerced* voters that can be influenced by the coercer; *votingWaitTime* and *helpRequestTime* that reflect the maximal number of steps the system waits for votes and notifications about being coerced, respectively. We denote such model by $\mathcal{M}(n, k, \mathit{maxCoerced}, \mathit{votingWaitTime}, \mathit{helpRequestTime})$.

In what follows, we omit auxiliary variables and actions that are not relevant to understanding the interplay between agents.

Agent *Environment*. The purpose of the *Environment* agent is twofold. Firstly, it serves as a container for variables shared by selected agents. The agents can have read-only or write-only access to the variables (denoted by *Can observe* and *Can set*, respectively, in agent interfaces in Figures 1, 2, and 3). Secondly, it traces the passage of time and changes the stage of elections. Namely, the elections start

Variables

- *vote*: 0. . . k
- *demandedVote*: 0. . . k

Actions

- $Vote_i$, for $i \in \{1, \ldots, k\}$
- $INeedVote_i$, for $i \in \{1, \ldots, k\}$
- *FetchGoodTracker*
- *CopyRealTracker*
- *Wait*
- *Finish*

Can observe

- WBB: *public election function*
- *elections' stage (init/voting/defense)*
- *his real tracker (when permitted)*
- *his exposed tracker*

Can set

- *his exposed tracker (via CopyRealTracker)*

Figure 1: A Voter agent

in the **initial stage**, when the secret bijection is non-deterministically prepared. Then, the **voting phase** is open and the clock is started. This phase ends when either all the voters send their choices or time exceeds *votingWaitTime*. Then, the system enters the **defense stage** and the clock restarts. The defense stage ends either when the clock exceeds *helpRequestTime* or all the voters execute the *Finish* action. Note that every agent can observe WBB, i.e., public election function, the stage, and the clock value. The clock limits are also public knowledge.

***Voter* agents.** Each *Voter* shares the same structure, presented in Figure 1. It is able to record via the *vote* variable the vote cast for choice $i \in \{1, \ldots, k\}$ by executing the action $Vote_i$. This action can be used only once, in the voting phase. It also records the coercer's request to vote for *demandedVote*. In both the cases 0 denotes that the variable is not set, i.e. the agent did not vote yet and has not been contacted by the coercer, respectively. In addition to the public variables of *Environment*, each *Voter* can observe his real tracker, obtained in the defense phase by executing action *FetchGoodTracker*. The agent can also observe his exposed tracker, i.e., the number assigned by *ElectionDS*, as presented to the *Coercer* agent. This becomes possible after requesting in the defense phase a tracker that points to a specific choice $i \in \{1, \ldots, k\}$, by firing action $INeedVote_i$. After obtaining his real tracker a *Voter* can decide to make it visible to the coercer by executing action *CopyRealTracker*. Finally, the agent can always *Wait*, unless the clock reaches the limit set for a phase. In the latter case, if it is the voting

Variables

- $\mathit{falseTrackerSentToVoter}_i$: Boolean, for $i \in \{1, \ldots, n\}$

Actions

- $\mathit{SetFalseTrackerOfVoter}_i\mathit{To}_j$, for $1 \leq i \leq n$, $1 \leq j \leq k$
- *Wait*

Can observe

- WBB: *public election function*
- *the secret bijection*
- *elections' stage (init/voting/defense)*

Can set

- *the exposed tracker of every Voter*

Figure 2: The *ElectionDS* agent

phase, then the *Voter* needs to decide on the vote immediately, and if it is the defense phase, then it automatically ends his participation by firing action *Finish*. It should be noted that these actions are autonomous, e.g., a *Voter* can signal *ElectionDS* that he is coerced to vote in a selected way, even if coercion does not take place.

Agent *ElectionDS*. The structure of *ElectionDS* agent is presented in Figure 2. The agent can, in addition to the public variables of *Environment*, observe the secret bijection function. This gives *ElectionDS* the full knowledge of the secret election function. The boolean variables $\mathit{falseTrackerSentToVoter}_i$ record that a voter $i \in \{1, \ldots, n\}$ requested and has been provided with a false tracker. This request is fulfilled by executing an action $\mathit{SetFalseTrackerOfVoter}_i\mathit{To}_j$ that sets the exposed tracker of voter $1 \leq i \leq n$ to choice $1 \leq j \leq k$. Note that while *ElectionDS* can set the value of the exposed tracker of any *Voter*, it cannot read the current value of the variable. Therefore, each *Voter* can first request a false tracker pointing to any choice and expose his real tracker afterwards, unknowingly to *ElectionDS*. Finally, *ElectionDS* can always *Wait*.

Agent *Coercer*. The structure of *Coercer* is presented in Figure 3. Starting from the initial phase until the votes are published, the agent can demand from any voter $1 \leq j \leq n$ to vote for $1 \leq i \leq k$, by executing $\mathit{ReqVote}_i\mathit{FromVoter}_j$ action. Such a request can be made at most once per voter and the total number of requests cannot exceed *maxCoerced*. These choices are recorded using variables $\mathit{voteDemandedFromVoter}_i$, where $1 \leq i \leq n$. As previously, the value of 0 signifies that no request has been made. The agent can observe all public variables of *Environment* and all the exposed trackers of all voters. At any step, the *Coercer* agent can *Wait*.

Variables

- $\mathit{voteDemandedFromVoter}_i : 0, \ldots, k$, for $i \in \{1, \ldots, n\}$

Actions

- $\mathit{ReqVote}_i\mathit{FromVoter}_j$, for $1 \leq i \leq k$, $1 \leq j \leq n$
- *Wait*

Can observe

- WBB: *public election function*
- *elections' stage (init/voting/defense)*
- *the exposed tracker of every Voter*

Figure 3: The *Coercer* agent

Atomic propositions. In order to construct formulae that can be interpreted in the model, we need some atomic propositions. We set $PV = \{\mathsf{finished}\} \cup \{\mathsf{vote}_{\mathsf{v,i}} \mid 1 \leq v \leq n, 1 \leq i \leq k\}$. Proposition finished denotes that the execution of the protocol has come to an end, and it holds iff all the voters have executed *Finish* or the clock has exceeded *helpRequestTime*. Formula $\mathsf{vote}_{\mathsf{v,i}}$ says that voter v has voted for candidate i; it holds iff v's variable *vote* contains i.

3.3 Implementation of the Model

We have used two different model checkers to verify properties of the model presented in Section 3.2. For exact model checking, we used MCMAS [15], which is the only publicly available tool for $\mathbf{ATL}_{\mathrm{ir}}$. MCMAS is based on the Interpreted Systems Programming Language (ISPL), which allows for higher-level descriptions of agents and their interaction. In Figure 4, we present the ISPL code implementing the *Coercer* agent. The local variables of the agent are denoted by *Vars*, *Lobsvars* denotes the set of the environment variables that the agent can observe, and *Actions* are action labels. The *protocol* section specifies which actions are available at what states; the *evolution* section defines the consequences of their execution. We refer to [15] for more details about MCMAS and ISPL. Unfortunately, exact model checking of abilities under imperfect information works only for very small models. To overcome this, we used the approximate model checking technique from [10]. We have developed a prototype tool implementing the technique, in which the explicit state variant of the model from Section 3.2 is hard-coded. The explicit state representation is completely isomorphic with the ISPL code. Both the ISPL code generator and the prototype tool are available online at `https://github.com/SeleneMC16/SeleneModelChecker`.

```
Agent Coercer

Lobsvars = {exposedTrackerOfVoter1, exposedTrackerOfVoter2};

Vars:
    coercedVoters: 0..2;
    voteDemandedFromVoter1: 0..2;
    voteDemandedFromVoter2: 0..2;
end Vars

Actions = {ReqVote1FromVoter1, ReqVote2FromVoter1,
           ReqVote1FromVoter2, ReqVote2FromVoter2, Wait};

Protocol:
    coercedVoters < maxCoerced and voteDemandedFromVoter1 = 0
    and Environment.votesPublished = false:
    {ReqVote1FromVoter1, ReqVote2FromVoter1, Wait};

    coercedVoters < maxCoerced and voteDemandedFromVoter2 = 0
    and Environment.votesPublished = false:
    {ReqVote1FromVoter2, ReqVote2FromVoter2, Wait};

    Other: {Wait};
end Protocol

Evolution:
    coercedVoters = coercedVoters + 1 if
    (Action = ReqVote1FromVoter1
    or Action = ReqVote2FromVoter1
    or Action = ReqVote1FromVoter2
    or Action = ReqVote2FromVoter2);

    voteDemandedFromVoter1 = 1 if Action = ReqVote1FromVoter1;
    voteDemandedFromVoter1 = 2 if Action = ReqVote2FromVoter1;
    voteDemandedFromVoter2 = 1 if Action = ReqVote1FromVoter2;
    voteDemandedFromVoter2 = 2 if Action = ReqVote2FromVoter2;
end Evolution

end Agent
```

Figure 4: ISPL code of the *Coercer* agent

4 Specification of Properties

In this section, we provide a list of example coercion-related specifications, formulated in the strategic logic $\mathbf{ATL}_{\mathrm{ir}}$. We reduce vulnerability to coercion to the ability of the coercer to learn about the value of the voter's vote. We also assume that the voter prefers to evade coercion, rather than cooperate with the coercer. Intuitively, this reflects the typical voters' attitude towards intimidation, rather than vote buying and bribery. We begin by a brief introduction to the logic.

4.1 Alternating-time Temporal Logic

Alternating-time temporal logic with imperfect information and imperfect recall ($\mathbf{ATL}_{\mathrm{ir}}$ [1,20]) generalizes the branching-time temporal logic **CTL** by replacing the path quantifiers E, A with *strategic modalities* $\langle\langle A\rangle\rangle$. Informally, $\langle\langle A\rangle\rangle\gamma$

expresses that the coalition A has a collective strategy to enforce the temporal property γ. The formulae make use of temporal operators: "X" ("next"), "G" ("always from now on"), "F" ("now or sometime in the future"), and U ("until").

Syntax. The language of $\mathbf{ATL}_{\mathrm{ir}}$ formulae is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\!\langle A \rangle\!\rangle \mathrm{X}\varphi \mid \langle\!\langle A \rangle\!\rangle \mathrm{G}\varphi \mid \langle\!\langle A \rangle\!\rangle \mathrm{F}\varphi \mid \langle\!\langle A \rangle\!\rangle \varphi \,\mathrm{U}\, \varphi,$$

where p stands for atomic propositions, and $A \subseteq \mathbb{A}\mathrm{gt}$ for any coalition of agents.

Strategies. A *strategy* of agent $a \in \mathbb{A}\mathrm{gt}$ is a conditional plan that specifies what a is going to do in every possible situation. Formally, it can be represented by a function $s_a : St \to Act$ satisfying $s_a(q) \in d_a(q)$ for each $q \in St$. Moreover, we require that $s_a(q) = s_a(q')$ whenever $q \sim_a q'$, i.e., strategies specify same choices in indistinguishable states. A *collective strategy* s_A for coalition $A \subseteq \mathbb{A}\mathrm{gt}$ is a tuple of individual strategies, one per agent from A. By $s_A[a]$ we denote the strategy of agent $a \in A$ selected from s_A.

Outcome paths. A *path* $\lambda = q_0q_1q_2\ldots$ is an infinite sequence of states such that there is a transition between each q_i, q_{i+1}. We use $\lambda[i]$ to denote the ith position on path λ (starting from $i = 0$). Function $out(q, s_A)$ returns the set of all paths that can result from the execution of strategy s_A from state q. Formally:

$out(q, s_A) = \{\lambda = q_0, q_1, q_2 \ldots \mid q_0 = q$ and for each $i = 0, 1, \ldots$ there exists $\langle \alpha^i_{a_1}, \ldots, \alpha^i_{a_k} \rangle$ such that $\alpha^i_a \in d_a(q_i)$ for every $a \in \mathbb{A}\mathrm{gt}$, and $\alpha^i_a = s_A[a](q_i)$ for every $a \in A$, and $q_{i+1} = o(q_i, \alpha^i_{a_1}, \ldots, \alpha^i_{a_k})\}$.

Function $out^{\mathrm{ir}}(q, s_A) = \bigcup_{a \in A} \bigcup_{q \sim_a q'} out(q', s_A)$ collects all the outcome paths that start from states indistinguishable from q to at least one agent in A.

Semantics. Let M be an *iCGS* and q its state. The semantics of **ATL** can be defined by the clauses below. We omit all the clauses for temporal operators except for "sometime", as they are not relevant for this paper.

- $M, q \models p$ iff $q \in V(p)$, and $M, q \models \neg\varphi$ iff $M, q \not\models \varphi$,
- $M, q \models \varphi \wedge \psi$ iff $M, q \models \varphi$ and $M, q \models \psi$, and $i \in \mathbb{N}$ we have $M, \lambda[i] \models \varphi$,
- $M, q \models \langle\!\langle A \rangle\!\rangle \mathrm{F}\varphi$ iff there exists a collective strategy s_A such that for all $\lambda \in out^{\mathrm{ir}}(q, s_A)$ there exists $i \in \mathbb{N}$ such that $M, \lambda[i] \models \varphi$.

In order to reason about the knowledge of agents, we add *modalities* K_a:

- $M, q \models \mathsf{K}_a\varphi$ iff $M, q' \models \varphi$ for all q such that $q \sim_a q'$.

That is, $\mathsf{K}_a\varphi$ says that φ holds in all the states that agent a considers possible at the current state of the world q.

4.2 Formulae for Coercion

Let us consider a coercer attempting to force a group of voters $A \subseteq \mathbb{A}\mathrm{gt}$ to vote for his preferred candidate. We can assume w.l.o.g. that the number of the candidate is 1. The formulae in Figure 5 express different "flavors" of the coercer's coercive ability, with the following reading:

$$\Phi_1 \quad \equiv \quad \langle\!\langle Coercer \rangle\!\rangle\, \mathrm{F}\, \big(\mathsf{finished} \wedge (\bigwedge_{v \in A} \neg\mathsf{vote}_{v,1} \rightarrow \mathsf{K}_{Coercer}(\bigvee_{v \in A} \neg\mathsf{vote}_{v,1})\big)$$

$$\Phi_2 \quad \equiv \quad \langle\!\langle Coercer \rangle\!\rangle\, \mathrm{F}\, \big(\mathsf{finished} \wedge (\bigvee_{v \in A} \neg\mathsf{vote}_{v,1} \rightarrow \bigvee_{v \in A} \mathsf{K}_{Coercer}(\neg\mathsf{vote}_{v,1})\big)$$

Figure 5: Formulae for model checking

configuration	#states	tgen	Lower approx.		Upper approx.		Approx.	Exact	
			tverif	result	tverif	result	result	(tg+tv)	result
$(2,1,1,1,1)$	427	<1	<1	False	<1	True	?	<1	True
$(2,1,1,4,4)$	16777	1	<1	False	<1	True	?	249	True
$(2,1,2,4,4)$	22365	1	<1	True	<1	True	True	timeout	
$(2,2,1,4,4)$	331441	19	1	False	16	True	?	timeout	
$(2,2,2,4,4)$	596577	36	2	True	31	True	True	timeout	
$(3,2,1,1,1)$	281968	40	<1	False	4	True	?	timeout	
$(4,1,1,1,1)$	146001	2	<1	False	2	True	?	timeout	
$(4,2,1,1,1)$	memout							timeout	

Figure 6: Experimental results for formula Φ_1

– Φ_1 expresses that the coercer can enforce a state where the elections are over and, if no one in A followed his orders, then the coercer knows that at least one of them disobeyed (but does not necessarily know who);
– Φ_2 says that if some of the voters did not vote as ordered, then the coercer will identify at least one of them.

Conceptually, the formulae capture the extent to which the coercer can identify the disobedience of the coerced voters, and hence knows when to execute his threats. Note that, when $A = \{v\}$, then both formulae are equivalent.

5 Verification

Selene is supposed to provide protection against coercion, so the formulae in Section 4 should in principle be all false. However, every protection mechanism has its limits. As noted in [19], even a single coercer can defeat the election defense system if the number of ballots is small or the coerced voter is particularly unlucky and the vote demanded by the coercer is not present in WBB. Also, the coercer's power intuitively increases with the number of voters he can simultaneously coerce. Finally, the exact limits of the coercer's ability to coerce become unclear when we consider more complex models, due to their combinatorial complexity. This is exactly when model checking can help to detect threats or verify correctness. In this section, we provide a preliminary attempt at model checking of the properties specified in Section 4.2 with respect to the models proposed in Section 3.2.

5.1 Exact and Approximate Model Checking of $\mathbf{ATL_{ir}}$

Synthesis and verification of strategies under partial observability is hard. More precisely, model checking of **ATL** variants with imperfect information has been proved $\mathbf{\Delta_2^P}$- to **PSPACE**-complete for agents that play memoryless strategies [9,20]. In our case, the following result applies.

Proposition 1 ([9,20]). *Model checking* $\mathbf{ATL}_{\mathrm{ir}}$ *is* $\mathbf{\Delta_2^P}$*-complete with respect to the number of the transitions in the model and length of the formula.*

The only publicly available tool for verification of imperfect information strategies is MCMAS [15], which essentially searches through the space of all the possible strategies. Nevertheless, no better algorithm is currently known, despite some recent attempts [6]. We employ MCMAS for *exact* model checking of our coercion specifications for Selene. An interesting alternative has been proposed recently in the form of *approximate model checking* based on fixpoint approximations of formulae [10]. The idea is to model check, instead of formula $\varphi \equiv \langle\langle A\rangle\rangle \mathrm{F}\mathsf{p}$, its upper and lower approximations $tr_U(\varphi)$ and $tr_L(\varphi)$:

- $tr_U(\varphi)$ verifies the existence of a *perfect information strategy* that achieves $\mathrm{F}\mathsf{p}$. Clearly, when there is no perfect information strategy to achieve it, then $\langle\langle A\rangle\rangle \mathrm{F}\mathsf{p}$ must also be false;
- $tr_L(\varphi)$ is a more sophisticated property, expressed in Alternating Epistemic μ-Calculus with Steadfast Next Step, with the property that the truth of $tr_L(\varphi)$ always implies φ. We refer the interested readers to [10] for details.

We have implemented the approximate algorithm from [10], together with a model generator for Selene, and ran a number of experiments with both exact and approximate model checking. The setup of the experiments, as well as the results, are presented in the rest of the section.

5.2 Experiments and Results

We collect the results of the evaluation for each of the specified formulae in tables presented in Figures 6 to 7. We show performance results for the approximation algorithms, both for the lower and the upper bound, and compare them to the exact verification done with MCMAS. Each row in a table corresponds to a single run of an experiment over the selected model. The columns contain the following information:

- the parameters of the model (*configuration*), consisting of the numbers of voters and available candidates, the maximal numbers of voters that the coercer can try to coerce and the clock steps that the system waits for incoming votes and for notifications from the voters about coercion attempts. E.g., configuration (2,2,2,4,4) describes the model with 2 voters, 2 candidates, the coercer coercing up to 2 voters, and the maximal time units for the system to wait for votes and coercion notifications being both set to 4;

configuration	#states	tgen	Lower approx.		Upper approx.		Approx.	Exact	
			tverif	result	tverif	result	result	(tg+tv)	result
$(2, 1, 1, 1, 1)$	427	<1	<1	False	<1	True	?	<1	True
$(2, 1, 1, 4, 4)$	16777	1	<1	False	1	True	?	257	True
$(2, 1, 2, 4, 4)$	22365	1	<1	True	<1	True	True	timeout	
$(2, 2, 1, 4, 4)$	331441	19	1	False	4	False	False	timeout	
$(2, 2, 2, 4, 4)$	596577	36	1	False	7	False	False	timeout	
$(3, 2, 1, 1, 1)$	281968	40	<1	False	1	False	False	timeout	
$(4, 1, 1, 1, 1)$	146001	2	<1	False	3	True	?	timeout	
$(4, 2, 1, 1, 1)$	memout							timeout	

Figure 7: Experimental results for formula Φ_2

- The size of the state space (*#states*) and the time that the algorithm spent on generating the data structures for the model (*tgen*);
- The running time and output of the verification algorithm (*tver, result*) for model checking the lower approximation $tr_L(\phi)$, and similarly for the upper approximation $tr_U(\phi)$;
- The result of the approximation (*Approx. result*), with "?" in case of inconclusive output;
- The total running time (*tg+tv*) and the result (*result*) of the exact $\mathbf{ATL}_{\mathrm{ir}}$ model checking with MCMAS.

The running times are given in seconds. *Timeout* indicates that the process did not terminate in 2 hours. *Memout* indicates that the process is terminated by the system due to allocating too much memory.

The exact $\mathbf{ATL}_{\mathrm{ir}}$ model checking is performed with MCMAS 1.3.0. To perform the approximate verification, we used the explicit representations of models from Section 3.2, and an implementation of the fixpoint algorithms from [10] in a stand-alone tool written in C++. The models used in both approaches were isomorphic. The tests were conducted on a Intel Core i7-6700 CPU with dynamic clock speed of 2.60 – 3.50 GHz, 32 GB RAM, running 64bit Ubuntu 16.04 Linux.

5.3 Discussion of the Results

As confirmed by the experiments, the question posed by formula Φ_2 is the most restrictive. Namely, in Φ_2 we ask whether the coercer has a general strategy to find out exactly which voter voted against his demands, assuming that there was a disobedient one. The answer to this question is true only in special cases of a single candidate. On the other hand, the results of verification of Φ_1 reveal that the system is sometimes not able to fully defend a coerced group and the coercer can detect that at least one of the members did not follow the demands. To illustrate this on a simple example, in a model of two voters and two ballots the coercer has a trivial strategy: request a vote for candidate 1 from both the voters. Moreover, in this case the coercer has even more knowledge, as he knows which one of the voters deceived (they both did).

$$\Phi_2^{dist_{A'}} \equiv \langle\langle Coercer \rangle\rangle\, \mathrm{F}\, \big(\mathsf{finished} \wedge ((\bigvee_{v\in A} \neg\mathsf{vote}_{v,1} \wedge \bigwedge_{v'\in Agents\setminus A} dist_{A'}(v')) \rightarrow \bigvee_{v\in A} \mathsf{K}_{Coercer}(\neg\mathsf{vote}_{v,1})\big)$$

Figure 8: A formula for quantitative analysis

Exact model checking with MCMAS seems infeasible in most of the cases, except for the small models up to hundreds of states. The approximations offer a dramatic speedup, enabling verification of models up to hundreds of thousands of states. Although the approximate method is faster, the results can be inconclusive; namely, we observe cases where the truth value of $tr_L(\varphi)$ differs from the value of $tr_U(\varphi)$. It should be noted that in the approximate approach the graphs are represented explicitely in memory, unlike in the case of BDD-based symbolic methods. Still, memory is cheaper and easier to buy than time.

5.4 Counting Coercion-Friendly Configurations

The validity of a property is a strong result: if a formula is true in the model, then the coercer has a strategy to achieve his goal under all possible circumstances. The system is therefore completely insecure against the considered type of attack. If, however, the formula turns out false, it does not mean that the system is always able to defend itself. In such case we only know that there is no uniform strategy that allows the coercer to break the system's defenses, given no information about the initial state of affairs (e.g., a partially uncovered choice function). We thus attempt to quantitatively estimate the extent to which our model is safe from the attacks expressed by Φ_2. To this end we inspect in detail all the possible distributions $D_{A'}$ of votes of voters outside of the coerced group A and check under which of these the coercer can precisely point to a disobedient voter. Formally, $dist_{A'} \in D_{A'}$ iff $dist_{A'}$ is a function from $Agents \setminus A$ to PV such that for each $v \in Agents \setminus A$ there exists $1 \leq i \leq k$ such that $dist_{A'}(v) = \mathsf{vote_{v,i}}$.

To perform quantitative analysis we utilise the formula $\Phi_2^{dist_{A'}}$ presented in Figure 8. Note that the formula depends on $dist_{A'} \in D_{A'}$ and $A \subseteq Agents$. Intuitively, it expresses the ability of the coercer to enforce that if some of the agents in A did not vote for candidate 1 and the remaining voters voted according to $dist_{A'}$, then the coercer can identify a voter in A that did not vote for 1.

The process of quantitative analysis is performed as follows. For a given model configuration, we fix an arbitrary coalition A. Then, for each distribution $dist_{A'} \in D_{A'}$ the formula $\Phi_2^{dist_{A'}}$ is verified. Our approach is based on state-labelling, hence we can inspect all the states reached just after publishing votes. In Fig. 9 by $\#vote$ we denote the aggregate number of such states that are consistent with any $dist_{A'} \in D_{A'}$ and $\#rvote$ collects the count of how many of these states satisfy $\Phi_2^{dist_{A'}}$. As we can observe, there are cases where the coercer can gain advantage in nearly half of considered distributions.

configuration	#states	tgen	result	#vote	#rvote	pvote $= \frac{\#rvote}{\#vote} \times 100\%$
$(2,2,1,4,4)$	331441	14	False	200	50	25%
$(2,2,2,1,1)$	9651	<1	False	144	36	25%
$(2,2,2,4,4)$	596577	24	False	360	90	25%
$(2,3,1,2,2)$	289423	10	False	378	168	44%
$(2,3,2,1,1)$	64829	1	False	576	256	44%
$(2,3,2,2,2)$	661501	22	False	864	384	44%
$(2,3,2,2,2)$	281968	15	False	672	84	12%
$(2,3,2,2,2)$	765232	41	False	1824	228	12%
$(4,1,1,1,1)$	146001	2	False	240	0	0%

Figure 9: Experimental results for formula $\Phi_2^{dist_{A'}}$ with percentage coverage

It should be emphasized that approximate algorithms are used for model checking $\Phi_2^{dist_{A'}}$. Thus, the *pvote* shows only the percentage of *confirmed cases* where a successful coercion strategy exists. The actual counts may be larger, since the approximations provide only guaranteed lower bound estimation.

6 Conclusions

In this paper, we present our first step towards model checking of e-voting protocols with respect to strategic abilities of their participants. We propose a simple multi-agent model of Selene, together with several formulae of $\mathbf{ATL}_{\mathrm{ir}}$ expressing coercion, and conduct preliminary experiments with model checking.

Our construction of the model is based on a natural pattern of dividing the outcome of an election into the public bulletin and the secret choice function, together with a secret bijection. We argue that the MAS approach provides a flexible framework for modelling security properties of voting protocols. In particular, it offers a natural separation of social and technical components and their interactions. Moreover, $\mathbf{ATL}_{\mathrm{ir}}$ offers a sensible trade-off between expressivity and veracity as the property specification language.

Model checking is done in two variants: exact, using MCMAS [15], and approximate, using the recently proposed methodology of fixpoint approximations [10]. As the experiments show, despite the prohibitive complexity of model checking with $\mathbf{ATL}_{\mathrm{ir}}$, the approximate method enables the analysis of many instances of our models, even in the presence of combinatorial explosion.

In the future, we plan to apply some recent developments in model reduction methods for strategic logics and allow for verification of more complex models. These include techniques such as abstraction, bisimulation-based reduction, and partial-order reduction. Moreover, we would like to extend the model of Selene with additional actors, such as coercers with conflicting goals.

Acknowledgements. The authors acknowledge the support of the National Centre for Research and Development (NCBR), Poland, under the PolLux project VoteVerif (POLLUX-IV/1/2016).

References

1. R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time Temporal Logic. *Journal of the ACM*, 49:672–713, 2002.
2. F. Belardinelli, R. Condurache, C. Dima, W. Jamroga, and A.V. Jones. Bisimulations for verification of strategic abilities with application to ThreeBallot voting protocol. In *Proc. of AAMAS*, pages 1286–1295. IFAAMAS, 2017.
3. J. Benaloh and D. Tuinstra. Receipt-free secret-ballot elections. In *Proc. of the 26th ann. ACM symp. on Theory of Computing*, pages 544–553. ACM, 1994.
4. I. Boureanu, P. Kouvaros, and A. Lomuscio. Verifying security properties in unbounded multiagent systems. In *Proceedings of AAMAS*, pages 1209–1217, 2016.
5. N. Bulling and W. Jamroga. Alternating epistemic mu-calculus. In *Proceedings of IJCAI-11*, pages 109–114, 2011.
6. S. Busard, C. Pecheur, H. Qu, and F. Raimondi. Reasoning about memoryless strategies under partial observability and unconditional fairness constraints. *Information and Computation*, 242:128–156, 2015.
7. S. Delaune, S. Kremer, and M. Ryan. Coercion-resistance and receipt-freeness in electronic voting. In *Computer Security Foundations Workshop, 2006. 19th IEEE*, pages 12–pp. IEEE, 2006.
8. J. Dreier, P. Lafourcade, and Y. Lakhnech. A formal taxonomy of privacy in voting protocols. In *Communications (ICC), 2012 IEEE International Conference on*, pages 6710–6715. IEEE, 2012.
9. W. Jamroga and J. Dix. Model checking ATL$_{ir}$ is indeed Δ_2^P-complete. In *Proceedings of EUMAS'06*, volume 223 of *CEUR Workshop Proc.* CEUR-WS.org, 2006.
10. W. Jamroga, M. Knapik, and D. Kurpiewski. Fixpoint approximation of strategic abilities under imperfect information. In *Proc. of AAMAS*, pages 1241–1249, 2017.
11. H. L. Jonker and W. Pieters. Receipt-freeness as a special case of anonymity in epistemic logic. *Proc. of the 19th Comp. Sec. Found. workshop*, pages 28–42, 2006.
12. R. Kusters and T. Truderung. An epistemic approach to coercion-resistance for electronic voting protocols. In *Security and Privacy*, pages 251–266, 2009.
13. R. Küsters, T. Truderung, and A. Vogt. A game-based definition of coercion-resistance and its applications. In *2010 23rd IEEE Computer Security Foundations Symposium*, pages 122–136. IEEE, 2010.
14. S. Lang. *Algebra.* Addison-Wesley, 1993.
15. A. Lomuscio, H. Qu, and F. Raimondi. MCMAS: An open-source model checker for the verification of multi-agent systems. *Int. Journal on Software Tools for Technology Transfer*, 2015. Available online.
16. Alessio Lomuscio and Wojciech Penczek. LDYIS: a framework for model checking security protocols. *Fundamenta Informaticae*, 85(1-4):359–375, 2008.
17. Bo Meng. A critical review of receipt-freeness and coercion-resistance. *Information Technology Journal*, 8(7):934–964, 2009.
18. T. Okamoto. Receipt-free electronic voting schemes for large scale elections. In *Security Protocols*, pages 25–35. Springer, 1998.
19. P.Y.A. Ryan, P.B. Rønne, and V. Iovino. Selene: Voting with transparent verifiability and coercion-mitigation. In *Proc. of Financial Cryptography and Data Security*, volume 9604 of *LNCS*, pages 176–192. Springer, 2016.
20. P. Y. Schobbens. Alternating-time logic with imperfect recall. *Electronic Notes in Theoretical Computer Science*, 85(2):82–93, 2004.
21. M. Tabatabaei, W. Jamroga, and P. Y. A. Ryan. Expressing receipt-freeness and coercion-resistance in logics of strategic ability: Preliminary attempt. In *Proc. of the PrAISe@ECAI Workshop*, pages 1:1–1:8. ACM, 2016.

An Internet Voting Protocol with Distributed Verification Receipt Generation

Kristjan Krips[1,4], Ivo Kubjas[2,4], and Jan Willemson[1,3]

[1] Cybernetica AS
Ülikooli 2, 51003 Tartu, Estonia
{kristjan.krips,jan.willemson}@cyber.ee
[2] Smartmatic-Cybernetica Centre of Excellence for Internet Voting
Ülikooli 2, 51003 Tartu, Estonia
ivo@ivotingcentre.ee
[3] Software Technology and Applications Competence Center
Ülikooli 2, 51003 Tartu, Estonia
[4] Institute of Computer Science, University of Tartu
Ülikooli 18, 50090 Tartu, Estonia

Abstract. This paper introduces an Internet voting scheme using a verification receipt design that splits the receipt generation between the voting client and vote collecting server. The aim of the protocol is to provide provable integrity properties even in the presence of a malicious component (most notably, client or server side malware). The protocol is designed to be used in low-coercion environments. We provide full description of the protocol, formally verifiable integrity proofs using EasyCrypt, and a discussion concerning other security requirements. The protocol was used in March 2016 for the Republican party caucus voting in the state of Utah, USA.

1 Introduction

The idea of Internet voting can be both appealing and terrifying. On one hand, it offers high accessibility to elections, arguably expected by the contemporary humans who are used to all kinds of online services. On the other hand, unlike in case of paper voting, all of its risks are not yet fully understood.

What the international community seems to agree upon is that a digital remote voting solution should provide clear verification and auditing mechanisms on both individual and system level. The exact nature of these mechanisms is determined by the threat model and available infrastructure.

In this paper we are going to present an Internet voting protocol that is meant to provide the voter with a post-election verification mechanism to detect vote manipulation attacks by both a malicious client application and voting server.

To achieve this, we propose a protocol design where the receipt generation is distributed between the two. An additional goal is formatting the receipt into the same data structure as the vote itself to utilize the existing functionality of multiple-choice ballots provided by the underlying technological platform.

Our design goals and choices have been motivated by the end user requirements. The protocol was designed to be used in March 2016 by the US Republican party for the option of online ballot submission during their caucus voting in the state of Utah.

As coercion was not regarded as a high risk in the environment, then no measures were implemented to allow for the voter to vote under coercion. Namely, given the receipt value, it is possible to uniquely determine the choice the voter voted for.

The protocol is built as an instance of TIVI platform[5], which in turn has foundations in the lessons learned while developing the Estonian Internet voting system. In particular, it shares several common characteristics (mix-net and provable decryption) with the IVXV protocol used in 2017 Estonian elections [16]. Still, the authentication mechanism was determined by the environment and the receipt generation routine was developed to match the verifiability properties required by the end user.

The voters had to pre-register to take part in online voting. Of 27490 pre-registered voters 24486 cast their vote using the online voting system [26].

To the best of our knowledge, this protocol has not yet been described in full detail in public sources (except for the JavaScript code that was implementing it). We believe that this description is of independent interest to the international community.

Our final contribution is modelling the protocol in EasyCrypt and providing formal verification of some target properties of the protocol. As integrity was stated as the most important goal by the end user, we concentrated on the integrity properties on both the client and server level.

2 Requirements set by the end user

Before designing an Internet voting protocol, the properties required from the protocol must be defined. The requirements depend on the specific environment and define the available methods for a particular event.

For example, coercion-resistance and post-election verification are somehow contradicting requirements – if the voter would be able to somehow verify that his/her particular vote was counted in the tally, then it would be possible to also prove to the coercer how the voter voted. Thus, there needs to be a clear decision between these two properties.

Thus, during the event preparation, the election organizers and Internet voting channel provider need to discuss and specify the expectations. Furthermore, the process needs to be transparent and possible risks should be made known to the parties and their possible effects estimated.

[5] `https://tivi.io/`

2.1 Integrity requirements

The main goal of the protocol is to provide an exact tally. To obtain this, we require that only eligible voters are able to cast a vote, votes can not be modified nor removed, and that the ballot is correctly encoded.

Requirement 1 (Eligibility). For every ballot in the final tally, there must exists a valid and registered voter who has cast a vote for a specific choice.

Requirement 2 (Tally integrity). After the voter has received a confirmation from the server that the submitted ballot has been stored, the stored ballot must be included in the final tally.

Requirement 3 (Ballot well-formedness). A ballot is a unique one-to-one representation of the voter's choice. The algorithms for encoding and decoding the vote are well-defined and correct.

2.2 Verifiability requirements

Due to environment settings, it was not possible to use pre-existing approach for individual verification using independent verification device [17] or return codes [4]. Hence, a slightly different concept of individual verifiability had to be defined.

The approach for individual verifiability was to have a receipt after casting a ballot which could be used for verifying that the ballot was correctly cast, stored and later tallied. The property is described as Requirement 4.

Furthermore, as the receipt was handed to the voter during the ballot casting, then the receipt had to include information which would allow for confidently claim any incorrect behaviour. The property is described as Requirement 5.

Finally, we consider the voter's privacy in this verifiability setting. We require that the cast ballot stays secret even for the election organizers unless the voter decides to challenge the receipt. On one side, this desists voters from applying illegitimate appeals. On the other side, this increases the trust in the appeal if it is successful, as different ballot content would be apparent. The property is described as Requirement 6.

Requirement 4 (Inclusion verifiability). Each voter can verify that her ballot has been included in the final tally.

As the final tally may be published long after the vote was cast, the information for verifying the ballot has to be stored. We will call both the stored information and the item it is stored on a *receipt*.

Requirement 5 (Liability provability). The receipt must include information which would allow to determine liability in the case when receipt verification fails.

Otherwise, it would be possible for a malicious voter (or a group of voters) to fake the receipts, claim that the tally is incorrectly computed and decrease the trust in the system. Eventually, this could even lead to Denial of Service when publishing results.

This requirement means that the receipt gives a strong proof of correctness. If the verification fails then either it is a case of election fraud, or a case of a dishonest voter or voting application. For the former it should be possible to prove the intent to cheat by election organizers, and for the latter it should be possible to prove the invalidity of the receipt.

A similar property, called "collection accountability" was described in [8].

We note that existence of verification receipts introduces the threat of coercion [13]. Hence, our protocol is only applicable in low-coercion environments.

Requirement 6 (Voter's privacy). Voter's choice must not become known to the election organizers unless the voter challenges the receipt.

2.3 Auditability requirements

The actions of election organizers must be auditable by an independent third party. More precisely, we require the following auditability properties.

Requirement 7 (Eligibility auditability). The auditor must be able to verify that only eligible voters have been able to vote and that no additional ballots have been added to the tally.

Requirement 8 (Decryption auditability). The auditor must be able to verify that the decryption operation is performed correctly on the encrypted ballots.

Requirement 9 (Privacy-preserving auditing). Auditing the election process must not threaten voters' privacy.

3 Protocol description

This section gives an overview of the protocol. We start by introducing the protocol participants and the used notation. Then we describe the assumptions that have to be fulfilled. Finally, we give an overview of the following four main phases of the protocol: distribution of keys, vote submission and receipt generation, vote decryption, post-election vote verification.

3.1 Participants

There are a number of parties involved in the protocol. As the primary party, we have Voter (VTR) as a physical participant. To participate in the elections, she needs the VotingClient (VC) software published by the election organizers. One voter is allowed to cast one vote and revoting is not possible.

There are several central functions that are under the control of the election organizers. However, there is a clear separation of duties.

First, there is the VotingServer (VS) that interacts with the VotingClient.

Secondly, the TallyServer (TS) decrypts the votes after the voting period ends. Due to Requirement 9, the ballots are shuffled using a mix-net before handing them over to TallyServer.

Thirdly, there is a KeyHolder (KH) that generates the encryption and signing keys, stores them securely and exports them at the correct protocol stage. The latter property is crucial to prevent revealing partial tally results. It can be implemented using different methods, e.g. hardware-based approach (using smart cards or HSMs) or distributed approach.

The read-append bulletin board (RABB) receives the cast ballots and stores them until the end of the election. The read-only bulletin board (ROBB) is initialized with a list of values and serves the values to the public.

Finally, there is a CertificationAuthority (CA) which provides confirmation of the voters' identities.

Correct operation of KH, CA and RABB are achieved by using the appropriate organizational measures and auditing. For example, the CA must conform to the standard requirements set to trust service providers, whereas the hash chain based RABB can be constantly monitored and its internal consistency can be verified by independent auditors.

3.2 Notation

The signature scheme that is used in the protocol is defined by three functions $(\mathsf{KGen}_{\mathsf{Sig}}, \mathsf{Sig}, \mathsf{Vf})$. The first of them is a key generation function that generates a key pair $(\mathsf{sk}, \mathsf{vk})$, which consists of a signing key and a verification key. The signing function Sig takes the signing key sk and some plaintext message pt, and returns the corresponding signature $\mathsf{s} = \mathsf{Sig}(\mathsf{sk}, \mathsf{pt})$. The verification function Vf takes the verification key vk, signature s and a message pt, and returns true if and only if the signature is obtained by using the corresponding signing key and plaintext. The function GetVer takes as input the signing key sk and returns the corresponding verification key vk.

In the implementation of our protocol, ECDSA was used as a signature scheme [18].

The encryption scheme is defined as a triplet of functions $(\mathsf{KGen}_{\mathsf{Enc}}, \mathsf{Enc}, \mathsf{Dec})$. The key generation function $\mathsf{KGen}_{\mathsf{Enc}}$ generates a pair of encryption and decryption keys $(\mathsf{ek}, \mathsf{dk})$. The encryption function Enc takes as input the encryption key ek, some random value ω and the message pt, and returns the ciphertext $\mathsf{ct} = \mathsf{Enc}(\mathsf{ek}, \omega, \mathsf{pt})$. The decryption function takes as inputs the decryption key dk and the ciphertext ct, and returns the corresponding plaintext $\mathsf{pt} = \mathsf{Dec}(\mathsf{dk}, \mathsf{ct})$. Additionally, there are functions Encode and Decode which map the voter's vote to an element of the selected representation set and back. Finally, there is a function Prove for providing the proof of correct decryption. If the decryption key dk is stored within a hardware module or is secret-shared, then only access to the functions $\mathsf{Dec}(\mathsf{dk}, \cdot)$ and $\mathsf{Prove}(\mathsf{dk}, \cdot, \cdot)$ are provided.

In the implementation of our protocol we are using ElGamal encryption scheme [15]. ElGamal works in a public group $\mathbb{G}$ of prime order p with generator g. The decryption key dk is used for finding the encryption key $\mathsf{ek} = g^{\mathsf{dk}}$. The encryption function Enc is defined as $\mathsf{Enc}(\mathsf{ek}, \omega, \mathsf{pt}) = (g^{\omega}, \mathsf{pt} \cdot \mathsf{ek}^{\omega})$ and the decryption function Dec is specified as $\mathsf{Dec}(\mathsf{dk}, \mathsf{ct}) = \mathsf{ct}_2 \cdot \mathsf{ct}_1^{-\mathsf{dk}}$.

ElGamal encryption scheme is homomorphic. Let us have two ciphertexts ct and ct' corresponding to the plaintexts pt and pt'. Then we have

$$\mathsf{ct} \cdot \mathsf{ct}' = (g^{\omega} \cdot g^{\omega'}, \mathsf{pt} \cdot \mathsf{ek}^{\omega} \cdot \mathsf{pt}' \cdot \mathsf{ek}^{\omega'}) = (g^{\omega+\omega'}, \mathsf{pt} \cdot \mathsf{pt}' \cdot \mathsf{ek}^{\omega+\omega'}),$$

the latter being an encryption of $\mathsf{pt} \cdot \mathsf{pt}'$.

The proof of correct decryption is denoted as $\mathsf{pf} = \mathsf{Prove}(\mathsf{dk}, \mathsf{pt}, \mathsf{ct})$. The proof is based on proof of discrete logarithm equality [1] and made non-interactive using Fiat-Shamir heuristic.

Ballot bt is a data structure representing the list $(\mathsf{ct}_1, \ldots, \mathsf{ct}_n)$ of ciphertexts, which typically are encrypted votes. In the described protocol, two-element ballots are used where the first element is an encrypted vote and second element represents the encrypted receipt. The function $\mathsf{Split}(\mathsf{bt})$ returns the corresponding ciphertexts as individually accessible elements $\mathsf{ct}_1, \ldots, \mathsf{ct}_n$.

For a receipt we will use the notation R, and an encrypted receipt will be denoted as rt.

In case a random value is needed, we use an appropriate space Ω to sample the value from.

3.3 Assumptions on the operating environment of the protocol

Assumption 1 (Existence of trust base). We assume that the participants belonging to the trust base behave according to the protocol and do not leak their private information.

The three trusted parties in the protocol are KeyHolder, RABB and CA. The protocol is designed such that the trust base would contain parties whose correctness can be audited using organizational measures and secured against external attacks using technical means.

Assumption 2 (Existence of a read-append bulletin board (RABB)). We assume the existence of a bulletin board which allows reading its current state and adding entries to the end of the bulletin board. Added entries are irreversible and unmodifiable, and the entries are strictly ordered with respect to the addition.

There have been several recent proposals to achieve such a primitive, based on e.g. threshold schemes [12] or Bitcoin-like block chain technology [23].

In our implementation, we are using hash chaining, where appending to the bulletin board is only allowed by the request signed by the voting server. The public interface allows queries for the chain elements using the chain index and the auditor interface allows retrieving a range of chain elements.

Assumption 3 (Existence of a read-only bulletin board (ROBB)). We assume the existence of a read-only bulletin board which is essentially a static key-value store that can be queried by the key.

In our deployment, the ROBB was implemented via a simple web-based query interface with a static CSV file containing the receipts and decrypted votes in the back-end. Note that no Private Information Retrieval mechanism was used, and hence a malicious ROBB could link verification IP addresses to the votes [9]. Additionally, the auditor could obtain the whole static CSV file.

The main difference between the ROBB and RABB is that the former is created in one action, but the latter allows continuous addition of items.

In our protocol, the two are used in different stages. The read-append bulletin board is used to commit the voting actions during the election period, whereas the read-only bulletin board is used for post-election verification queries.

Assumption 4 (Existence of voter certification). We assume that the election organizer has prior knowledge of every eligible voter and that there is an independent certification authority which provides confirmation of their identities.

This assumption can be implemented in several ways depending on the specific environment. Some jurisdictions may have a constantly updated voter registry, some may rely on the voters registering themselves at the election organizers some time before the elections are set up. In our application scenario, the latter was the case. Since the voting period was only one day, the voters' list was kept static throughout that period.

Assumption 5 (Existence of pre-channel to voters). We assume the existence of an authenticated and secure pre-channel between every voter and the election organizer.

The pre-channel could be initialized using a national PKI if it exists. In the alternate case other methods like email and SMS channel could be used. It is possible to add additional technical measures to increase the privacy of the channel. For example, it is possible to use email over encrypted channel, mutually authenticating the servers and signing emails using DKIM [20].

3.4 Assumptions on the attack model

To simplify the modelling of the protocol in EasyCrypt, we need to define additional assumptions which restrict the capabilities of the adversaries.

Assumption 6 (Perfectly binding signature scheme). We define an abstract signature scheme that is assumed to be perfectly binding. This means that for any generated signature s on a message pt there does not exist another message pt' such that verification succeeds.

Otherwise, it would be possible to construct another message for which some signature is valid. This assumption is required for achieving auditability and post-election verifiability.

Assumption 7 (Perfectly hiding encryption scheme). We assume that the defined encryption scheme is perfectly hiding. This means that given a ciphertext ct on a message pt, no adversary can learn the encrypted message.

In practice, this assumption does not hold. For example, in case of ElGamal encryption scheme, the naïve success probability of polynomial-time adversary would be $\frac{1}{|\Omega|}$. The success probability could be increased by using systematic attacks against the encryption scheme.

However, as currently known solutions for breaking ElGamal security over safe group parameters give the adversary only negligible success probability, then for the ease of modelling the protocol, we use the aforementioned assumption.

Assumption 8 (No cooperation between adversarial parties). We assume that no two adversarial parties cooperate.

As currently the approach for proving the security of the protocol is to prove the security of the protocol for different adversaries, then allowing cooperating adversaries would exponentially grow the amount of required proofs. Thus, this assumption keeps the number of different security proofs manageable.

As a result, the corresponding security claims do not hold in the case where malicious VotingClient and malicious VotingServer cooperate.

Assumption 9 (Perfect randomness). We assume that if any honest party queries for a random value, then the value is sampled from a uniform distribution.

The assumption ensures that the probability of colliding values between different parties in the protocol is negligible.

3.5 Protocol phases

To simplify the presentation of the protocol, we will divide it into four interconnected subprotocols. The first subprotocol defines the distribution of keys between different parties. The second subprotocol describes the voting process and voting receipt generation. The third subprotocol describes how votes are decrypted and published on the ROBB. Finally, the fourth subprotocol describes vote verification by the voter.

Distribution of keys: The key generation and distribution of the encryption key pair is done by KeyHolder. Before the election period starts, it generates the key pair $(\mathsf{ek}, \mathsf{dk})$ and publishes the public part ek to the VotingClient and VotingServer. After the voting period has closed, it sends the private part dk of the key to the TallyServer.

The CertificationAuthority generates its own signing key pair $(\mathsf{sk}_{\mathrm{CA}}, \mathsf{vk}_{\mathrm{CA}})$ and distributes the verification key $\mathsf{vk}_{\mathrm{CA}}$ to the VotingServer and TallyServer.

Every voter needs her own key for signing the ballot. Thus, a signing key pair $(\mathsf{sk}_{\mathrm{VTR}}, \mathsf{vk}_{\mathrm{VTR}})$ has to be generated for every voter. To prove the identities of the voters, CertificationAuthority signs the corresponding verification key to obtain the certificate c_{VTR}. The signing keys are delivered to the voters over an encrypted and authenticated channel. The certification authority stores only the voter's verification key $\mathsf{vk}_{\mathrm{VTR}}$ and certificate c_{VTR}.

VotingServer generates a signing key pair $(\mathsf{sk}_{\mathrm{VS}}, \mathsf{vk}_{\mathrm{VS}})$ that is used for signing the receipts. The public part $\mathsf{vk}_{\mathrm{VS}}$ is sent to the VotingClient to allow for verifying the validity of the receipts.

Vote submission and receipt generation: In order to satisfy Requirement 4, we need to provide the voter with a way to verify that the ballot actually has been included in the final tally. Note that the voter was not required to obtain cast-as-intended assurance during the online protocol phase. Rather the requirements stated in Section 2 aim at generating a receipt that can be later used for post-election verification. Essentially, the receipt will be the query key to request the voter's decrypted choice from the ROBB.

There are two parties that could generate the receipt – VotingClient and VotingServer. However, a malicious VotingServer or several collaborating instances of VotingClient could then manipulate the votes and generate receipts that would leave the voter with impression that everything is fine (see below in this Section for more detailed attack descriptions).

Hence, we decided to introduce a design where both the VotingClient and VotingServer would be generating a part of the receipt. As the protocol will be using a mix-net and provable decryption based on ElGamal encryption, our goal was to also make the receipt to be an ElGamal cryptogram so that it would go through the mix-net and would allow for a decryption proof.

These goals are achieved by our vote submission and receipt generation protocol in Figure 1. The VotingClient encodes voter's choice as a plaintext pt and encrypts it to get the ElGamal cryptogram ct.

To form the receipt, the VotingClient and VotingServer will generate random exponents r_1 and r_2, respectively. The receipt to be used during the verification will be $\mathsf{R} = g^{r_1+r_2}$. However, there are still potential attacks to be taken into account while doing so.

If the VotingServer is malicious then by knowing the input r_1 from the VotingClient, it could provide its part r_2 in the receipt in a way that the verifying voter would accept the receipt even if a wrong ballot was counted in the tally. For mitigation, we encrypt the VotingClient's part of the receipt as rt_1 and only send this to the VotingServer. As our aim is to produce an ElGamal-encrypted receipt rt anyway, sending a homomorphic partial encryption is sufficient. The VotingServer then deterministically encrypts its value r_2 to obtain rt_2 and combines the encrypted parts to obtain the encryption $\mathsf{rt}' = \mathsf{rt}_1 \cdot \mathsf{rt}_2$.

On the other hand, a malicious set of VotingClient instances could cooperate during the receipt generation. The instances could construct a choice-receipt database, where for every possible choice there is a valid receipt R. In this case, the voter would be presented with R, although the VotingClient has cast a ballot for another choice with different receipt R', which is then discarded. Thus, verifying the vote by the voter would incorrectly succeed during post-election verification period. To mitigate this attack, we need to provide input from the VotingServer such that VotingClient could not modify its part. Thus, VotingServer signs the combined information rt' (which should equal rt) to prevent modifications by VotingClient. The resulting protocol is depicted in Figure 1.

The VotingServer checks that the voter has not already cast a vote before forwarding the ballot to the RABB. After submitting the ballot, it is stored on the RABB and the VotingClient is sent the storage index. The protocol is illustrated in Figure 2.

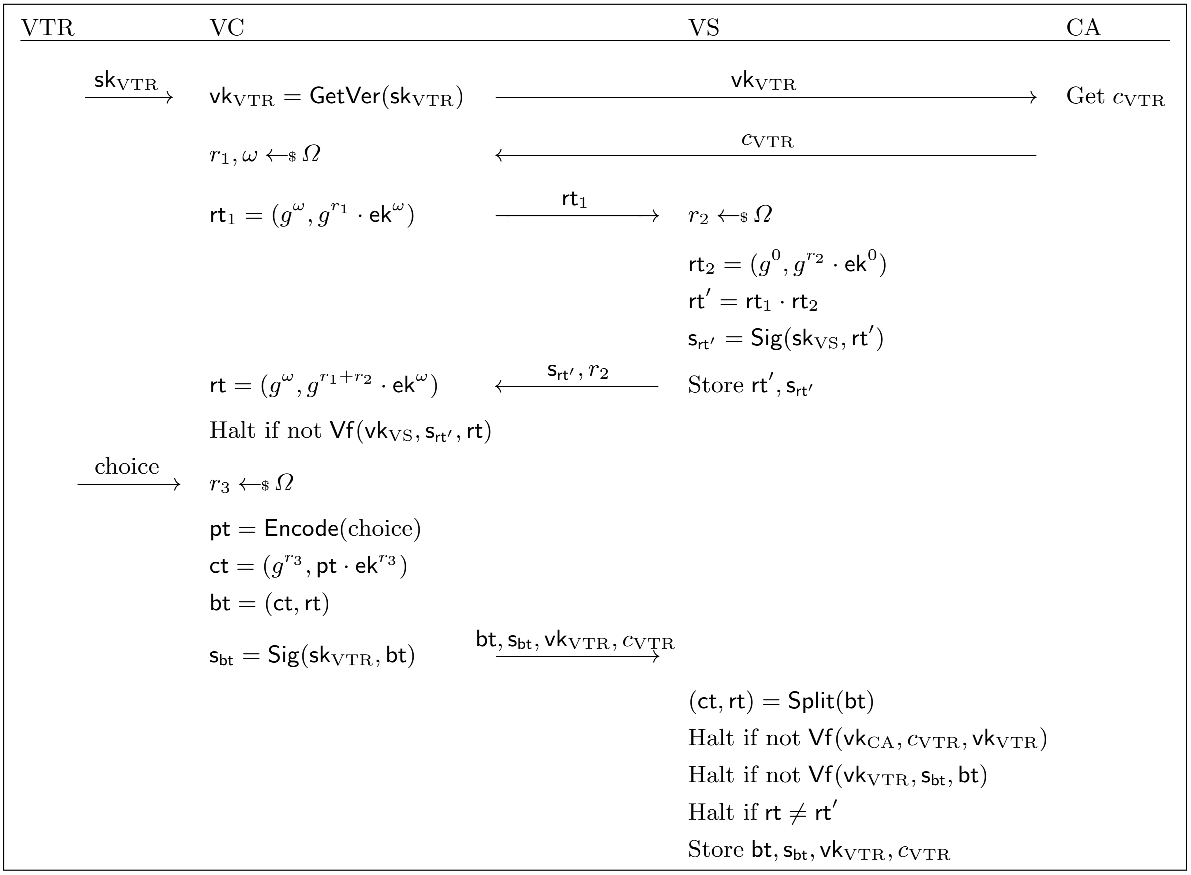

VTR
VC
VS
CA
$\xrightarrow{\mathsf{sk}_{\mathrm{VTR}}}$
$\mathsf{vk}_{\mathrm{VTR}} = \mathsf{GetVer}(\mathsf{sk}_{\mathrm{VTR}})$
$\xrightarrow{\mathsf{vk}_{\mathrm{VTR}}}$
Get c_{VTR}
$r_1, \omega \leftarrow_{\$} \Omega$
$\xleftarrow{c_{\mathrm{VTR}}}$
$\mathsf{rt}_1 = (g^{\omega}, g^{r_1} \cdot \mathsf{ek}^{\omega})$
$\xrightarrow{\mathsf{rt}_1}$
$r_2 \leftarrow_{\$} \Omega$
$\mathsf{rt}_2 = (g^0, g^{r_2} \cdot \mathsf{ek}^0)$
$\mathsf{rt}' = \mathsf{rt}_1 \cdot \mathsf{rt}_2$
$\mathsf{s}_{\mathsf{rt}'} = \mathsf{Sig}(\mathsf{sk}_{\mathrm{VS}}, \mathsf{rt}')$
$\mathsf{rt} = (g^{\omega}, g^{r_1+r_2} \cdot \mathsf{ek}^{\omega})$
$\xleftarrow{\mathsf{s}_{\mathsf{rt}'}, r_2}$
Store $\mathsf{rt}', \mathsf{s}_{\mathsf{rt}'}$
Halt if not $\mathsf{Vf}(\mathsf{vk}_{\mathrm{VS}}, \mathsf{s}_{\mathsf{rt}'}, \mathsf{rt})$
$\xrightarrow{\text{choice}}$
$r_3 \leftarrow_{\$} \Omega$
$\mathsf{pt} = \mathsf{Encode}(\text{choice})$
$\mathsf{ct} = (g^{r_3}, \mathsf{pt} \cdot \mathsf{ek}^{r_3})$
$\mathsf{bt} = (\mathsf{ct}, \mathsf{rt})$
$\mathsf{s}_{\mathsf{bt}} = \mathsf{Sig}(\mathsf{sk}_{\mathrm{VTR}}, \mathsf{bt})$
$\xrightarrow{\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}}$
$(\mathsf{ct}, \mathsf{rt}) = \mathsf{Split}(\mathsf{bt})$
Halt if not $\mathsf{Vf}(\mathsf{vk}_{\mathrm{CA}}, c_{\mathrm{VTR}}, \mathsf{vk}_{\mathrm{VTR}})$
Halt if not $\mathsf{Vf}(\mathsf{vk}_{\mathrm{VTR}}, \mathsf{s}_{\mathsf{bt}}, \mathsf{bt})$
Halt if $\mathsf{rt} \neq \mathsf{rt}'$
Store $\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}$

Fig. 1. Vote submission and receipt generation protocol.

As noted above, the receipt R is the value $g^{r_1+r_2}$ where r_1 is generated by the VotingClient and r_2 is generated by the VotingServer. For verification, R is encoded using ASCII printable characters and displayed to the voter. For auditing, the tuple $(i, \mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, r_1, r_2, \mathsf{s}_{\mathsf{rt}'})$ is provided to the voter in the form of a QR-code.

Vote decryption: Once the voting period has ended, the ballots are stripped of identifying information and sent to the TallyServer. To further protect privacy during auditing, the ballots are shuffled using a mix-net and then provably decrypted. In that particular election, a mix-net based on a proof of a shuffle by Terelius and Wikström [27] was used.

The decryption process and result publishing are shown in Figure 3.

Post-election vote verification: The voter can use any modern web browser to query the ROBB by the receipt R to obtain the choice and compare the result to her cast choice. We note that for providing information-theoretic privacy

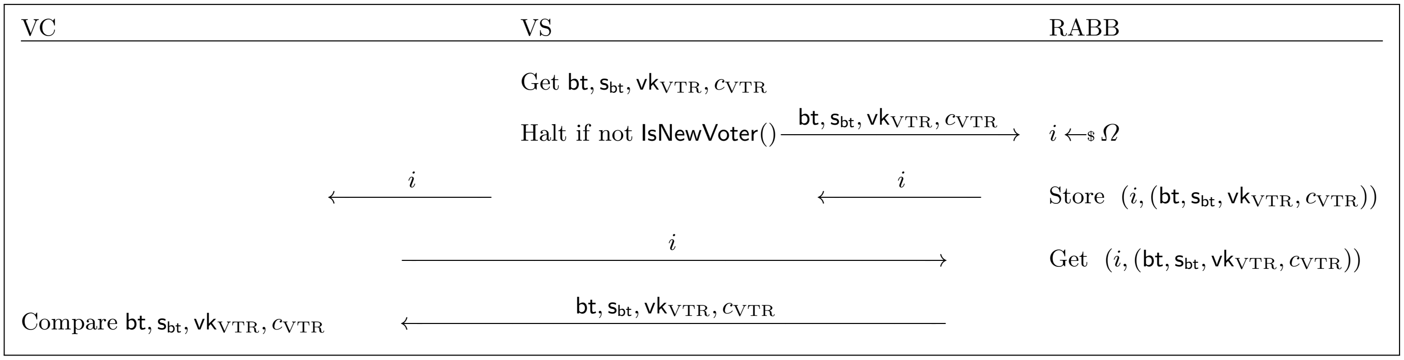

VC
VS
RABB
Get $\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}$
Halt if not $\mathsf{IsNewVoter}()$
$\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}$
$i \leftarrow_{\$} \Omega$
i
i
Store $(i, (\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}))$
i
Get $(i, (\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}))$
$\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}$
Compare $\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}$

Fig. 2. Ballot storage protocol

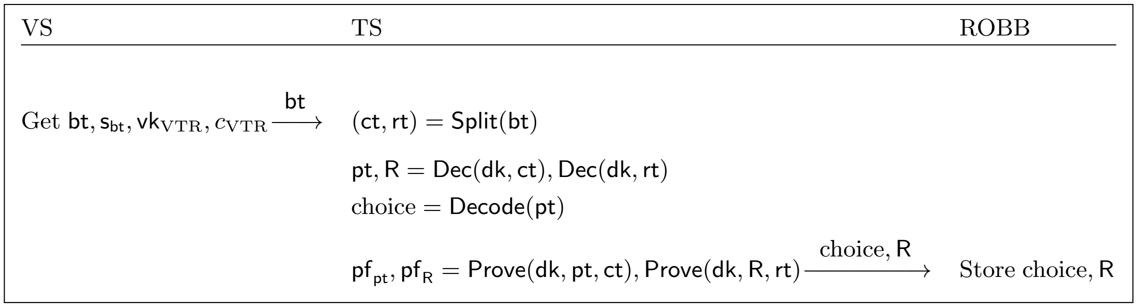

VS
TS
ROBB
Get $\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\mathrm{VTR}}, c_{\mathrm{VTR}}$
bt
$(\mathsf{ct}, \mathsf{rt}) = \mathsf{Split}(\mathsf{bt})$
$\mathsf{pt}, \mathsf{R} = \mathsf{Dec}(\mathsf{dk}, \mathsf{ct}), \mathsf{Dec}(\mathsf{dk}, \mathsf{rt})$
$\text{choice} = \mathsf{Decode}(\mathsf{pt})$
$\mathsf{pf}_{\mathsf{pt}}, \mathsf{pf}_{\mathsf{R}} = \mathsf{Prove}(\mathsf{dk}, \mathsf{pt}, \mathsf{ct}), \mathsf{Prove}(\mathsf{dk}, \mathsf{R}, \mathsf{rt})$
choice, R
Store choice, R

Fig. 3. Ballot decryption and publishing protocol

of the vote, we would need to transmit the whole content of ROBB to the VotingClient [9].

4 Requirement implementation and verification

Requirement 1 (Eligibility). This requirement is fulfilled by organizational measures. The trusted CertificationAuthority makes sure that only eligible voters are given access to the signing keys, and auditors verify the signed votes on RABB to make sure that only the votes of eligible voters are on the bulletin board.

Requirement 2 (Tally integrity). Tally integrity is assured by auditing the entries on the bulletin boards. Due to the properties of the bulletin boards, it is not possible to remove entries, and all the cast ballots must be stored there.

All of the tallied votes are published on the ROBB along with the receipts. Voters are able to use any Internet connected device that contains a browser to query the ROBB and check if their vote was tallied. Thus, if the ballot was not transferred to the TallyServer or not tallied, the voter would detect the absence of the ballot. On the other hand, if a server would have acted maliciously by not sending a vote to the bulletin board, the VotingClient would have detected this during the ballot storage protocol.

Requirement 3 (Ballot well-formedness). The encoding function Encode takes the ASCII-encoded choice, considers its byte form and interprets it as an integer. If the length of the choice is fixed, there exists an ElGamal group with large enough subgroup such that there is an injective mapping. The decoding function Decode applies the inverse operations. If the decoding function outputs an error message on an incorrectly encoded ballot, the requirement is satisfied.

Requirement 4 (Inclusion verifiability). While casting the vote the VotingClient and VotingServer generate a receipt and the VotingClient displays it to the voter. Additional cryptographic information used to generate the receipt is given to the voter in the form of a QR-code.

The encrypted receipt is added to the ballot and sent to the voting system. The tallied vote with the corresponding receipt is published in the ROBB. The voter can use her receipt to query ROBB to check if her vote was tallied. In case of a mismatch or a missing vote the voter can appeal by using the info on the stored QR-code.

Requirement 5 (Liability provability). According to the receipt storage protocol, the voter would have access to $\mathsf{R}, (i, \mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, r_1, r_2, \mathsf{s}_{\mathsf{rt}'})$. The voter obtains the pair $(\text{choice}, \mathsf{R})$ from the ROBB using R as the key. If the value choice differs from the voter's choice, then the voter may appeal the result.

During the appeal process, the following steps are performed together by the election organizer and voter to detect the liable party:

1. The values $\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\text{VTR}}, c_{\text{VTR}}$ are fetched from RABB using i. The fetched values bt and s_{bt} are compared to the corresponding values on the receipt.
2. The voter's signing key is verified using vk_{CA}, c_{VTR} and vk_{VTR}.
3. The ballot signature is verified using vk_{VTR}, s_{bt} and bt.
4. The ballot is split into ciphertexts $\mathsf{Split}(\mathsf{bt}) = (\mathsf{ct}, \mathsf{rt})$.
5. The signature on the receipt is verified using vk_{VS}, $\mathsf{s}_{\mathsf{rt}'}$ and rt.
6. The receipt is decrypted as $\mathsf{R}' = \mathsf{Dec}(\mathsf{dk}, \mathsf{rt})$. The receipt is compared to $g^{r_1+r_2}$.
7. The ciphertext is decrypted as $\mathsf{pt} = \mathsf{Dec}(\mathsf{dk}, \mathsf{ct})$ and decoded as $\text{choice} = \mathsf{Decode}(\mathsf{pt})$. The choice is compared to one stored on ROBB.

We consider how every step allows to determine the cheating party:

1. As illustrated in Figure 2, the VotingClient obtains the values $\mathsf{bt}, \mathsf{s}_{\mathsf{bt}}, \mathsf{vk}_{\text{VTR}}, c_{\text{VTR}}$ from RABB using the index i during ballot storage.
   If the step fails, then either VotingClient skipped obtaining the values from RABB or one of VotingClient and voter must have modified the receipt.
2. As illustrated in Figure 1, VotingServer halts if $\mathsf{Vf}(\mathsf{vk}_{\text{CA}}, c_{\text{VTR}}, \mathsf{vk}_{\text{VTR}})$ fails. Thus, if the same operation fails during appeal then the election organizer cheated.

3. Similarly to previous step, the values have been verified during vote submission stage and if the step fails, then the election organizer has been cheating.
4. The ballot was split during the vote submission stage and if the step fails, then the VotingServer has behaved incorrectly. Thus, the election organizer cheated.
5. The signature is also verified by the VotingClient during ballot submission and if the step fails, then the signature $\mathsf{s}_{\mathsf{rt}'}$ is invalid on the receipt. This means that either the VotingClient or voter have modified the receipt.
6. As the value r_1 is provided by the VotingClient and it verified the correctness of the signature $\mathsf{s}_{\mathsf{rt}'}$ during the vote submission stage, then the failed comparison indicates that either VotingClient or voter modified the values r_1 or r_2 on the receipt.
7. If the comparison fails, then the result of decrypting the same ciphertext ct twice is inconsistent. It is possible to verify the proof $\mathsf{pf}_{\mathsf{pt}}$ to confirm that the election organizer has cheated.

In conclusion, if the Steps 2, 3, 4 or 7 fail, the election organizer cheats. If the Steps 1, 5 or 6 fail, then either VotingClient or the voter is cheating.

Requirement 6 (Voter's privacy). The KeyHolder sends the decryption key to TallyServer after the election period has ended. Furthermore, the encrypted ballots are stripped of identifying information and additionally shuffled. Thus, there is no preliminary access to the encrypted ballots, and after the election the voter privacy is preserved.

In the previous description we saw that in case the voter challenges the receipt, then her ballot is decrypted and the privacy is void.

Requirement 7 (Eligibility auditability). The auditor has a view of the ballots stored on the RABB and the ballots sent to the TallyServer for decryption. He checks that the signature s_{bt} for every ballot bt verifies using $\mathsf{vk}_{\mathrm{VTR}}$, and that the certificate c_{VTR} for $\mathsf{vk}_{\mathrm{VTR}}$ verifies using $\mathsf{vk}_{\mathrm{CA}}$. He then checks that the ballots sent for decryption correspond to the ballots stored on RABB.

Requirement 8 (Decryption auditability). As defined in Subsection 3.2, ElGamal encryption scheme is used. ElGamal decryption can be implemented in a provable way [1]. The auditor can retrieve the proofs $\mathsf{pf}_{\mathsf{pt}}$ and pf_{R} of correct decryption generated during the protocol illustrated in Figure 3. As the auditor can verify these proofs, this requirement is satisfied.

Requirement 9 (Privacy-preserving auditing). Before transmitting the ballots to the TallyServer, a shuffling mix-net is applied. The auditor sees shuffled ballots and the corresponding decryptions, but can not map shuffled ballots to unshuffled ballots, thus keeping the privacy of the voters.

5 Modelling the protocol with EasyCrypt

5.1 Formal verification

Creating secure cryptographic primitives is difficult. When primitives are combined to create protocols, the complexity increases even further, which may introduce subtle vulnerabilities. A classic example is the Needham–Schroeder protocol, where a severe vulnerability was found 15 years after the original paper was published [22,7].

One way to avoid such problems is to state formal security criteria and use computing tools to verify them. Unfortunately, such tools still have a long way to go before they can be used to verify all the desired properties of the protocols. Hence in practice we usually take a combined approach, modelling the protocol, formally checking as many properties as possible, and presenting heuristic arguments for the others. This paper follows the same path.

This far, majority of the formal proof attempts for cryptographic protocols have been made in the symbolic model, using the ProVerif software suite (e.g. [19], [14], [3]). Computational model is arguably closer to reality, but also more complex. This is why the tools supporting the computational approach (like EasyCrypt [5]) have matured more recently. Nevertheless, first proofs of privacy properties have already been given using EasyCrypt by Cortier *et al.* [11]. In this paper, we will be using EasyCrypt to model the protocol and to check a few integrity and verifiability properties.

EasyCrypt uses a probabilistic While language for modelling the primitives and protocols [6]. The use of an imperative language supports the choice of modelling the protocol with EasyCrypt.

5.2 Formal model of the protocol

Before modelling the voting protocol we had to decide which abstraction level to use. A detailed protocol description would allow to verify the details of the desired security properties. However, proving the security of implementation details could be complex or even impossible with EasyCrypt. For example, the current EasyCrypt version does not have a framework to support mix-nets and therefore we did not include mix-nets in our protocol model [11]. Hence, in order to focus on the desired security claims we decided to abstract away several details when describing the protocol in EasyCrypt. Thus, we modelled the protocol run with one voter that did not include mix-net and did not include provable decryption. The resulting EasyCrypt code is posted to a GitHub repository.[6]

After modelling the protocol we did a sanity check to make sure that the protocol runs as intended. We refer to the EasyCrypt lemma `votingCorrectness` to show that vote verification succeeds with probability 1 in case all parties follow the protocol. Then we showed that a malicious VotingClient who randomly changes the vote of the voter will remain undetected with probability $\frac{1}{q}$, where q is the number of candidates. This holds in case the voter uses the receipt for verification. Thus, the malicious client is not detected by vote verification when the randomly sampled vote matches the vote that was cast by the voter. Finally, we tested that neither the VotingServer nor TallyServer are able to remove votes without getting caught. This was formalized with the lemmas `voteDeletingVServer` and `voteDeletingTally`. For more details about

[6] `https://github.com/krips/uvoting`

the formalization of the lemmas, see the linked EasyCrypt source code in the gist repository.

5.3 Clash attack

While modelling the receipt protocol, we found that in the described form it was susceptible to a clash attack. Clash attack allows the malicious voting software to modify votes by reusing receipts such that different voters who vote for the same candidate get the same receipt [21]. The current voting protocol was designed to avoid clash attacks, but the problem occurs due to the way how the user interaction is implemented. Namely, the receipt is given to the voter *after* she has already made the choice in the voting software.

Thus, the malicious VotingClient could first collect the receipts of different voters and then check if a receipt for the voter's current choice is available. In case the receipt for the current choice is available, the malicious software could change the vote and display a receipt that corresponds to a vote that was cast by another voter.

A proof of concept adversarial voting client was observed during the event [2]. The site notified the user against the dangers of Internet voting and did not perform a coordinated attack.

We modelled the proof of concept attack in EasyCrypt and showed that in case of two voters who vote for the same candidate the malicious VotingClient can change the vote of the second voter by reusing the receipt that was given to the first voter. The description of the attack is given in the attached EasyCrypt code, more specifically in the module `ReceiptForgery` and in lemma named `clashAttack`.

The problem can be resolved by slightly modifying the way how the messages are shown in the user interface of the VotingClient. The voter should receive the receipt before she enters the vote to the VotingClient as then the malicious software has to pick the receipt before the voter has selected the candidate.

5.4 Experiences with EasyCrypt

We learned quite a lot by using EasyCrypt. In order to share the experience, we will discuss some of the observations and hope that they are valuable to the developers and other users of the tool.

Proving security in the computational model gives more precise results compared to the symbolic model, but the additional price to pay is increased complexity. EasyCrypt allows to specify complex algebraic protocols relatively easily, but proving the desired security properties is not straightforward. When applying a proof tactic, multiple subgoals may appear and all of them have to be proven in order to move on with the proof. It is often the case that proving simple subgoals takes more time than proving the security aspects.

Goals can be proven in EasyCrypt either by manually combining existing proof tactics or by using an automated SMT solver Alt-Ergo [10]. A subgoal may seem trivial, but if the SMT solver is not able to prove it, one has to start looking through the extensive library of theory files to find lemmas and axioms that can be used to close the subgoal. One way to simplify this process is to use the built-in *search* command which helps to find all lemmas and axioms that correspond to a given pattern. For example, if we would need to find properties about accessing map elements, then we could use the command *search*(_.[_]), which would return all lemmas and axioms that use the corresponding operator.

As noted above, specifying the voting protocol in EasyCrypt language was relatively easy. The main question was on how to model communication between the parties and for that we created additional functions for the parties and a separate module which modelled the execution of the protocol with a single voter. The first difficulties emerged when we had to choose which of the existing theories to use. EasyCrypt is a work in progress which means that its theories and documentation are changing. For some theories there were multiple versions available in order to provide backward compatibility. Thus, we came across a situation where we had to choose if we would like to use a new theory or an old one to model our protocol. Both options seemed to work, but we decided to use the older version as it had helpful proof examples.

The next difficulty appeared when proving a simple property about the success probability of a malicious VotingClient. It appeared that a specific version of the *seq* tactic was not documented in the reference manual, and thus we had to find other means to understand how the probability parameters could be used in connection with the *seq* tactic.

We found a small code dependency issue when setting up a fresh EasyCrypt install from the main branch. Namely, two weeks after starting to model the protocol it appeared that an old EasyCrypt theory had been removed from the main branch. Thus, we had to either rewrite some of the code or to pin EasyCrypt to a specific commit. Rewriting the code was easy, but modifying the underlying modules will usually break the proofs.

This brings us to the second lesson. We had to expand the protocol description once some of the proofs were already finished, and this meant that a large part of the proofs had to be updated. Changes to the underlying modules may break the proofs due to change of invariants, shifting of line numbers and added conditions. Thus, updating proofs might not be difficult, but it can still be time consuming when multiple proofs have to be modified.

6 Related work

The novelty of our protocol comes from the way how the receipts are generated. The receipt generation requires the voting client and server to work together and check that the other party would adhere to the protocol. If the voter uses the receipt to verify the vote, she can be sure that both the voting client and the server did not deviate from the receipt generation protocol. In case the receipt verification fails, it is possible to determine which of the two parties acted maliciously.

There are multiple voting schemes which use receipts, but the most similar to our scheme is the Selene voting protocol created by Peter Ryan *et al.* [25]. Selene aims to provide transparent verifiability and coercion-mitigation at the same time. It uses tracking numbers for voters, posted to a public bulletin board along with the votes. Thus, a voter can take her tracking number and check from the public bulletin board if the vote corresponding to the tracking number is the one that was cast. Pedersen commitments are used for generating a trapdoor for the tracking numbers. This allows the voter to use the trapdoor to lie to a coercer. In addition, to prevent coercion, the tracking numbers are revealed after the vote has already been cast.

Comparing Selene to our protocol, we clearly see that Selene was designed to be coercion resistant while our protocol was designed for elections where coercion is not an issue. However, resolving disputes involving the receipts is non-trivial in Selene, while in our protocol shared cryptographic information helps to reveal who is to blame.

When the voter is not participating in the receipt generation process, it is much harder to find out the misbehaving party. This is illustrated in the paper by Küsters *et al.* [21], which described how it is possible to attack the verifiability of multiple e-voting systems by reusing the receipts. The paper showed that this kind of an attack was possible against ThreeBallot and VAV [24], but also against a version of Helios [1] using aliases. However, if the voter and the voting server jointly generate the receipt as in our scheme, clash attacks can be mitigated by displaying the receipt value to the voter before the voter makes her choice. After the receipt value has been shown to the voter, the attacker is not able to use someone else's vote and receipt to replace the vote without getting caught.

7 Conclusion

In this paper we presented a new Internet voting protocol that supports post-election vote verification using a receipt that is jointly generated by mutually distrusting voting application and voting server. We showed how the proposed cryptographic scheme in conjunction with organizational measures can be used to meet all the required integrity, verifiability and auditability requirements. In addition, we modelled the protocol in EasyCrypt and formally proved some of the integrity properties.

The protocol was used to run a Republican caucus vote in the state of Utah in March 2016. Altogether, 24486 votes were cast using our system. A certain amount of politically emotional discussions concerning security issues was, of course, expected, but this far the discussion has been lacking technical details and precise security claims. We hope that our paper will help to fill some gaps and clear some of the misunderstandings.

Acknowledgements

The research leading to these results has received funding from the Estonian Research Council under Institutional Research Grant IUT27-1 and the European Regional Development Fund through the Estonian Centre of Excellence in ICT Research (EXCITE) and the grant number EU48684.

References

1. Ben Adida. Helios: Web-based Open-Audit Voting. In *USENIX security symposium*, volume 17, pages 335–348, 2008.
2. Andrew Appel. Internet Voting, Utah GOP Primary Election. `https://freedom-to-tinker.com/2016/03/22/internet-voting-utah-gop-primary-election`, 2016.
3. Michael Backes, Catalin Hritcu, and Matteo Maffei. Automated verification of remote electronic voting protocols in the applied pi-calculus. In *CSF'08*, pages 195–209. IEEE, 2008.
4. Jordi Barrat, Ben Goldsmith, David Jandura, John Turner, and Rakesh Sharma. Internet voting and individual verifiability: the Norwegian return codes. *Electronic Voting*, pages 274–283, 2012.
5. Gilles Barthe, Benjamin Grégoire, Sylvain Heraud, and Santiago Zanella Béguelin. Computer-aided security proofs for the working cryptographer. In *CRYPTO'11*, volume 6841 of *LNCS*, pages 71–90. Springer, 2011.
6. Gilles Barthe, Benjamin Grégoire, César Kunz, Yassine Lakhnech, and Santiago Zanella Béguelin. Automation in computer-aided cryptography: Proofs, attacks and designs. In *Certified Programs and Proofs*, pages 7–8, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
7. R. K. Bauer, T. A. Berson, and R. J. Feiertag. A key distribution protocol using event markers. *ACM Trans. Comput. Syst.*, 1(3):249–255, August 1983.
8. M. Bernhard, J. Benaloh, J. A. Halderman, R. L. Rivest, P. Y. A. Ryan, P. B. Stark, V. Teague, P. L. Vora, and D. S. Wallach. Public Evidence from Secret Ballots. *ArXiv e-prints*, July 2017.
9. B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 41–50, Oct 1995.
10. Sylvain Conchon, Évelyne Contejean, Johannes Kanig, and Stéphane Lescuyer. CC(X): Semantical combination of congruence closure with solvable theories. In *Post-proceedings of SMT 2007*, volume 198(2) of *Electronic Notes in Computer Science*, pages 51–69. Elsevier Science Publishers, 2008.
11. Véronique Cortier, Benedikt Schmidt, Constantin Catalin Dragan, Pierre-Yves Strub, Francois Dupressoir, and Bogdan Warinschi. Machine-checked proofs of privacy for electronic voting protocols. In *IEEE S&P'17*. IEEE Computer Society Press, 2017.
12. Chris Culnane and Steve Schneider. A peered bulletin board for robust use in verifiable voting systems. In *Computer Security Foundations Symposium (CSF), 2014 IEEE 27th*, pages 169–183. IEEE, 2014.
13. Stéphanie Delaune, Steve Kremer, and Mark D. Ryan. Coercion-resistance and receipt-freeness in electronic voting. In *19th IEEE Computer Security Foundations Workshop (CSFW'06)*, pages 28–42, 2006.

14. Stéphanie Delaune, Steve Kremer, and Mark D. Ryan. Verifying properties of electronic voting protocols. In *Proceedings of the IAVoSS Workshop On Trustworthy Elections (WOTE'06)*, pages 45–52, June 2006.
15. Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4):469–472, 1985.
16. Sven Heiberg, Tarvi Martens, Priit Vinkel, and Jan Willemson. Improving the Verifiability of the Estonian Internet Voting Scheme. In *E-Vote-ID*, volume 10141 of *LNCS*. Springer.
17. Sven Heiberg and Jan Willemson. Verifiable Internet Voting in Estonia. In *6th International Conference on Electronic Voting: Verifying the Vote, EVOTE 2014, Lochau / Bregenz, Austria, October 29-31, 2014*, pages 1–8, 2014.
18. Cameron F. Kerry and Patrick D. Gallagher. Digital Signature Standard (DSS), 2013.
19. Steve Kremer and Mark Ryan. Analysis of an electronic voting protocol in the applied pi calculus. In *ETAPS 2005*, volume 3444 of *LNCS*, pages 186–200. Springer, 2005.
20. Murray Kucherawy, Dave Crocker, and Tony Hansen. DomainKeys Identified Mail (DKIM) Signatures. RFC 6376, September 2011.
21. Ralf Kusters, Tomasz Truderung, and Andreas Vogt. Clash Attacks on the Verifiability of E-Voting Systems. In *IEEE S&P'12*, pages 395–409. IEEE Computer Society, 2012.
22. Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Commun. ACM*, 21(12):993–999, December 1978.
23. Pierre Noizat. Blockchain electronic vote. In David Lee Kuo Chuen, editor, *Handbook of Digital Currency*. Elsevier, 2015. Chapter 22.
24. Ronald L. Rivest and Warren D. Smith. Three Voting Protocols: ThreeBallot, VAV, and Twin. In *Proceedings of USENIX/ACCURATE Electronic Voting Technology (EVT)*, 2007.
25. Peter Y. A. Ryan, Peter B. Rønne, and Vincenzo Iovino. Selene: Voting with transparent verifiability and coercion-mitigation. In *Financial Cryptography and Data Security*, volume 9604 of *LNCS*, pages 176–192. Springer, 2016.
26. Smartmatic. Utah Republican Party 2016 Preference Caucus - case study. `http://www.smartmatic.com/uploads/tx_news/CS_UTAH_2016_ENG.pdf`, 2016.
27. Björn Terelius and Douglas Wikström. Proofs of restricted shuffles. In *AFRICACRYPT 2010*, volume 6055 of *LNCS*, pages 100–113. Springer, 2010.

Electryo, In-person Voting with Transparent Voter Verifiability and Eligibility Verifiability

Peter B Rønne$^{[0000-0002-2785-8301]}$, Peter Y A Ryan, Marie-Laure Zollinger

University of Luxembourg
Esch-sur-Alzette, Luxembourg
{peter.roenne,peter.ryan,marie-laure.zollinger}@uni.lu

Abstract. Selene is an e-voting protocol that allows voters to directly check their individual vote, in cleartext, in the final tally via a tracker system, while providing good coercion mitigation. This is in contrast to conventional, end-to-end verifiable schemes in which the voter verifies the presence of an encryption of her vote on the bulletin board. The Selene mechanism can be applied to many e-voting schemes, but here we present an application to the polling station context, resulting in a voter-verifiable electronic tally with a paper audit trail. The system uses a smartcard-based public key system to provide the individual verification and universal eligibility verifiability. The paper record contains an encrypted link to the voter's identity, requiring stronger assumptions on ballot privacy than normal paper voting, but with the benefit of providing good auditability and dispute resolution as well as supporting (comparison) risk limiting audits.

1 Introduction

In this paper, we propose combining the highly transparent counted-as-intended verification mechanism of the e-voting scheme Selene [26] with paper ballot, in-person voting. The aim is to keep the vote casting experience close to paper ballot voting with optical scanning, while enabling the intuitive voter-verification of the Selene scheme. The resulting scheme provides improved dispute resolution and supports risk limiting audits.

For most end-to-end verifiable schemes the voter verifies the presence of an encryption of her vote in the input to the tally on the bulletin board. In contrast, Selene lets a voter check that her vote appears correctly, in the clear, in the final tally via a tracking number system. This provides a highly transparent and intuitive verification, but, if naively implemented, could lead to vote-selling and coercion. The main idea of Selene is to mitigate the coercion threats by notifying the voters of their tracking number only after the full list of tracking numbers and votes has been published. Coerced voters can then simply choose a tracker pointing to the required vote and claim it as theirs. The notification provides the voter high assurance that it is the correct, i.e. unique, tracker while being deniable in the event of coercion.

In a paper ballot election the voters enjoy ballot secrecy thanks to the isolation of the voting booth at the polling station - giving good resistance against coercion and vote-buying attempts. Normally, but with UK as a prominent counter-example, the ballots are also anonymous and unmarked, extending the ballot secrecy to the tally phase. However, the integrity of the election relies on trust assumptions for the talliers, and many real attacks and errors are known, as shown in [16]. In Germany, for example, the tally process is public [3] and at least gives the voters the possibility to oversee the tally ceremony, however considerable trust is still required in the chain of custody of ballots.

To improve on this situation we propose here introducing the Selene mechanism to allow voters to verify that their vote is counted-as-intended. This requires the introduction of a carefully protected link between the ballot and the voter. The vote casting experience of the system is close to the optical scan paper ballot systems with the difference that the paper ballots will have a (QR-)code printed onto them which contains an encryption of the voter's identity.[1] We assume that voters have smartcards to store and prove their ID. Before getting into the details we recall the key elements of Selene.

1.1 The Essence of Selene

Selene revisits the old idea of enabling verification by posting the votes in the clear on the BB along with a private tracking numbers. The new twist is that voters are only notified of their tracker some time after the vote/tracker pairs have been publicly posted, giving a coerced voter the opportunity to choose an alternative tracker that will placate the coercer. Notification of the trackers is carefully designed to provide assurance that it is the correctly assigned tracker, i.e. unique to the voter, while being deniable. The key goals of Selene are:

- Ensure that each voter is assigned a unique tracker number.
- Notify the voter of her tracker after the vote/tracker pairs have been published in a manner that provides high assurance and yet is deniable in the event of coercion.

This is achieved, in essence, by publishing a list of trackers, n_i, verifiably encrypting and shuffling these and assigning them to the voters under trapdoor commitments according to the secret permutation π resulting from the shuffles. The commitment for the ith voter takes the form:

$$C_i := \mathsf{pk}_i^{r_i} \cdot g^{n_{\pi(i)}}$$

Where pk_i is the voter's public trapdoor key. C_i is a Pedersen commitment to the tracker but can also be thought of as the second term (β) of an exponential ElGamal encryption of the tracker under the ith voter's trapdoor public key pk_i. The corresponding first term ($\alpha = g^{r_i}$) is not published, but is communicated to the voter over a private channel at notification time. On receipt of the α-term, the voter can combine this with the β-term and decrypt using her trapdoor key.

[1] This might be troublesome in some jurisdictions.

If she is coerced, she can choose an alternative tracker that will satisfy the coercer and compute, using her trapdoor key, an alternative α. Without the trapdoor, it is intractable to compute an α that will decrypt to a given tracker. This observation simultaneously underpins the assurance that the tracker is correct, and removes the need to authenticate the α as communicated to the voter.

1.2 The Essence of Electryo

The key innovation of Electryo is to introduce a protected link between the paper ballot and the voter ID, in such a way as to guarantee the integrity and the secrecy of the link. This link is used to associate the encrypted vote, scanned from the paper ballot, with the voter ID on the BB, thus enabling the Selene mechanism to kick in. An additional feature is that at the time of scanning the ballot, a fresh, random *receipt code* is generated and printed for the voter to retain. This is required later to access the tracker number, providing an extra layer of privacy, as explained in detail later.

Now that voters are able to verify their vote in the clear, we can omit the usual checks required in cryptographic, end-to-end verifiable schemes: Benaloh challenges and correct posting to the *BB* of the encrypted vote. A corollary of this last observation is that the voter does not need to retain a copy of the encrypted vote, just the receipt code, which helps ensure receipt-freeness.

The voting system provides individual verifiability via the Selene check, allows universal verifiability of the setup phase and of the tally as well as eligibility via the digital signatures. The paper record provides a basis for dispute resolution, while risk-limiting audits will strengthen the link between the paper and digital record – all of this while preserving a good measure of coercion-mitigation.

The outline of the paper is as follows: Below we give a brief overview of related work. In section 2 we list the parties involved, as well as the primitives used. Section 3 will give details of the voting ceremony from the voter's point of view. In section 4, we will give further details of the scheme including the cryptography. Section 5 gives a brief analysis of the scheme, describing some potential attacks and counter-measures.

Related Work Several in-person voting protocols mix paper ballots or a paper-audit trail with a public digital record of the votes:

Prêt-à-Voter [25] is a paper-based voting scheme with voter-verifiability, a version of which has been trialled in a state election [13]. Contrary to the present scheme, these schemes does not provide transparent verification or directly support RLAs, see however [20] for a version with a human-readable paper-audit trail.

Wombat [6] combines paper-ballot voting with cryptographic tabulation and end-to-end verifiability. A voting machine delivers a paper ballot containing a plaintext vote as well as the encrypted version as a QR-code. The voter can check the correctness of the plaintext vote before putting it in a ballot box. The encrypted version is scanned and posted to a *BB* and the paper copy is kept by the voter as a receipt.

Another polling station e-voting scheme is STAR-Vote [5] which combines electronic voting machines (DREs) with a paper trail to achieve end-to-end verifiability and allow for efficient risk limiting audits (RLAs). The correctness of the encryption of the vote, can be tested by the voter by a sort of Benaloh challenge, where discarded ballots are decrypted in public. We will give a more detailed comparison in the long version of this paper (to appear). Note that it was not a design goal of STAR-vote to have eligibility verifiability.

All of these schemes do not provide the transparent verification of the plaintext vote in the final tally, as we do here.

We also note that there are other schemes based on trackers, specifically sElect [18] and the boardroom scheme analysed in [4], but they are not paper-based and do not provide the level of coercion-resistance and receipt-freeness that we aim for here.

2 Participants and Primitives

The main participants of the protocols are:

- The voters V_i. We assume that they are provided with electronic ID cards[2], e.g. as part of a national electronic ID infrastructure like in Estonia [1].[3] The card stores a secret signing key together with the ID which is associated with the corresponding public verification key vk_i. We assume that the card can perform an encryption of the ID with the election key and sign input using the secret signing key. Further the voter has public and secret key pair, $\mathsf{pk}_i, \mathsf{sk}_i$ for the Selene mechanism, where the latter is stored in a Vote Supporting Device (VSD) e.g. a smartphone or a computer and perhaps also on the smartcard.
- The Election Authority EA is managing the election and protocol setup.
- The Tally Tellers TT create the public election key PK_T and threshold share the secret key. They also facilitate a Mixnet M which is used to ensure privacy, and performs parallel verifiable re-encryption mixes see e.g. [27].
- A public web Bulletin Board BB is used for verifiable communication, and will be assumed to be append-only and have a consistent public view.
- The Tracker Retrieval Authority (TRA) is responsible for relaying communication between the voters and TT. Specifically TRA will send the so-called α term to the voter, which can be turned into a tracker for her vote using the secret key sk_i.
- Registration Clerks and Talliers assisting at the polling station.
- Printers with Card Readers. These print a ballot code, bcode, onto the paper ballots in the form of a QR-code, containing the re-encrypted ID and digital signature of the voter.

[2] See [2] for an example of a smartcard implementing ElGamal encryption with Elliptic Curves.

[3] See [21] for a recently found flaw in that system, demonstrating the importance of a secure implementation of this system.

- Optical Scanners with a Receipt Printer. The Scanner reads out the voter's choice on the paper-ballot and the bcode, and sends an encryption of the vote to BB together with a re-encryption of the ballot code and an encrypted receipt code. It delivers a *ballot proof* to the voter, that contains the receipt code in plain text together with a digital signature for accountability.

Some primitives used are

- Encryption. We assume an IND-CPA secure homomorphic encryption scheme allowing re-encryption and verifiable mixing. To be explicit we choose ElGamal encryption which was used in Selene, and the homomorphic properties are needed for the Selene mechanism. We denote encryption with the key PK $\{\cdot\}_{PK}$ and re-encryption is denoted $\{\cdot\}'_{PK}$. For some parts the homomorphic properties are not necessary and we use a RCCA secure scheme instead, i.e. the only malleability of the ciphertext is the ability to re-encrypt which is necessary for privacy and mixnets. To be explicit we can use the OAEP 3-round transformation [23, 22] of ElGamal. A single ciphertext then basically consists of two ElGamal ciphertexts and is RCCA secure under the Gap Diffie-Hellman assumption. We denote this encryption $\{\cdot\}_{\text{OAEP},PK}$. The parallel mixing is easily adapted to this encryption scheme since it basically consists of two ElGamal ciphertexts.
- Zero-Knowledge Proofs. We use zero-knowledge proofs, and proofs of knowledge, as well as signatures to ensure universal verifiability. For non-malleability the strong form [11] of the Fiat-Shamir transform [15] is used for obtaining non-interactive proofs, and we further include election identifiers in the hash to avoid malleability across elections.
- Plaintext equivalence tests (PETs). A PET [17] produces a public verifiable test whether two ciphertexts are an encryption of a same plaintext message, without revealing the plaintexts to anybody. The test requires a threshold set of the Tellers TT.
- QR-codes. A QR-code is a matrix barcode containing information for reliable and easy scanning. The encryption schemes used here can be based on elliptic curves requiring in the order of 512 bit strings. A ciphertext could then e.g. be stored in a QR-code version 6 (up to 1088 bits) or a version 10 for two OAEP ciphertexts.

3 Voting Experience

In this section we describe the protocol from the voter's perspective. The vote-casting ceremony is close to a paper-ballot election with optical scanning of the ballots. The entire voting experience is described in figure 1 and more cryptographic details will be given in next section.

3.1 Registration

We assume that all voters are in possession of ID smart-cards, e.g. as part of a national electronic ID infrastructure. The ID card can create signatures and

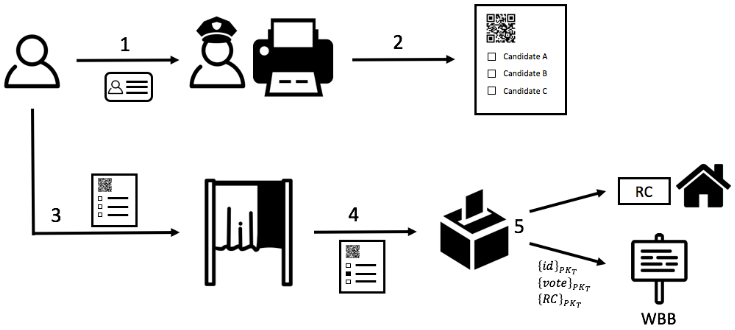


Fig. 1: Description of the voting phase.
(1) The voter enters the polling station and goes to a registration clerk with her ID card to be identified. (2) Her ID card is read and the printer delivers the ballot with the encrypted ID contained in a QR-Code. (3) The voter goes to a booth to fill her ballot. (4) She puts her ballot into a ballot box containing the scanner, (5) that sends the encrypted vote to the bulletin board and prints the voter a take home receipt code.

will be used to authenticate voters. The registration of the voters could probably happen automatically if based on a national PKI, alternatively by a company etc.

Besides the ID-card, the Selene mechanism assumes that each voter V_i holds a secret key, sk_i. This may require a registration step by the voters, the details of which we omit, but note that these keys could potentially be used for multiple elections. The tracker authority TRA also needs to know where to contact the voter for the tracker retrieval phase, e.g. an email address. Such confirmed contact data is normal to have in an electronic ID infrastructure. However, for improved usability, we assume that the voter is using an app (e.g. authenticated via the sim card as in Estonia [1]) that accesses sk_i e.g. via the smartcard or, if properly authenticated, the program could be the creator of the Selene keys.

3.2 Voting phase

On the voting day, the voter presents her ID card to a poll worker to confirm ID and eligibility, as in standard elections the voters showing up can be recorded in a paper log. The printer is equipped with a smart card reader and interacts with the card to retrieve an encryption, by the smart-card, of the ID and signature. It prints an unfilled ballot with a QR-code which encodes a re-encryption of the voter ID and digital signature, confirming the voter was present.

Then the voter enters the booth to fill the ballot, and finally she heads to a ballot box with a scanner/printer. The latter delivers a receipt code RC_i on paper, without releasing it, before scanning the ballot. The scanner re-encrypts the ballot code, encrypts the vote and releases the receipt code to the voter. The data is stored and sent to BB after voting ends, and the paper ballot is retained in the ballot box.

3.3 Tracker retrieval

After the tally phase, cast votes and corresponding tracking numbers will appear on the BB. After a pause, allowing coerced voters to access this information, the voters will receive their α-term (see introduction) via their support device at randomised times, as in Selene. The device will calculate the voter's unique tracker using the received α-term, the public β-term and the trapdoor key sk.

However, in contrast to Selene the α term will only be sent to the voter if she at some point after election enters a correct receipt code RC_i in her device. This adds a layer of privacy as explained later in section 5.3.

3.4 Voting in case of coercion

Coerced voters can take steps to mitigate the coercion. After the tally board is created with votes and corresponding trackers, the voter can choose a tracker pointing to a candidate of the coercer's choice. Further, the voter can calculate a fake α-term using sk that opens to this tracker. The voter can now show the coercer this tracker and α-term, if required.

Further, for improved coercion-resistance, the coerced voter can also contact TRA with authentication and request to not receive the real α-term, but only the fake. Now, even in the case where a coercer or vote buyer controls the interface to receive the α-term, he does not receive any convincing evidence of the cast vote. As mentioned above the essential assumption here is that the voter has access to sk, e.g. via multiple copies or the storage on the ID card.

In a longer version of this paper (to appear), we will present an alternative version of Selene where the coerced voter even before or during voting can contact TRA and request a faked α-term.

3.5 Comments on usability

The voting experience is close to a standard optical-scan scheme. As with an optical-scan or STAR-Vote [5], the scanner and ballot box can be combined, so that the ballot will be read before being fed in automatically in the ballot box. The only aspect that might be a bit troubling for some voters is the printing of the QR code on the ballot form. This does not affect usability, it is automatic as far as the voter is concerned, but might be worrying from a privacy perspective.

We avoid the verification steps such as Benaloh challenges [7] of the encryptions. Instead, we have the extra Selene verification phase with the receipt code and tracker check, which we believe is more understandable for voters.

For disabled persons, multi-lingual communities or generally complicated ballots, voting machines could also be used to fill out the ballots. Here the QR code created by the printer is scanned by the voting machine to produce the filled out ballot, which is kept as a paper record. A scanning step is not necessary in this case.

4 Protocol Description

We now describe the protocol in more technical detail including cryptography.

4.1 Pre-Election Setup

Let us recall Selene's set-up phase that we will also follow here [26].

The Tally Tellers set up a secure group and create the threshold election key PK_T for ElGamal encryption (or another homomorphic encryption scheme). We assume that all voters have PKs in the chosen group. Let $\mathsf{pk}_i = g^{x_i}$ be the public key of voter V_i, and $x_i = \mathsf{sk}_i$ their secret key. The Election Authority publishes on BB the set of tracking numbers n_i. These could just be $1, \ldots, n$ with n the number of eligible voters. Using a verifiable re-encryption mix each voter is associated a unique secret encrypted tracker on BB: $(\mathsf{ID}_i, \{g^{n_j}\}_{PK_T})$, where $j := \pi(i)$, and π is the secret permutation resulting from the mixes. As described in detail in [26], the Tally Tellers $TT_1, \ldots, TT_t$ produce a trapdoor commitment $C_i = \mathsf{pk}_i^{r_i} \cdot g^{n_j}$ where $r_i = \sum_{k=1}^{t} r_{i,k}$, along with an α-term $\alpha_i = g^{r_i}$ that will be kept secret under encryption. Only TT_k knows $g^{r_{i,k}}$.[4]
Before vote casting BB displays

$$(\mathsf{ID}_i, \mathsf{vk}_i, \mathsf{pk}_i, \{g^{n_j}\}'_{PK_T}, C_i)$$

Here vk_i is the verification key for voter V_i, and the corresponding secret key is stored along with ID_i on the voter's smartcard. The smartcard can produce signatures that can be verified via vk_i and we assume the signature scheme to be existentially unforgeable. Further, the smartcard can produce encryptions that can be used in the mixnet construction and decrypted by TT. We here use ElGamal encryption $\{\cdot\}_{PK_T}$, the OAEP version thereof discussed above, and e.g. Schnorr signatures, but other choices are possible, and since the smartcards are used across elections it might be preferable to use a separate key for this part.

4.2 Voting

Voter V_i goes to the polling station and is identified and registered by a clerk. If her identity is confirmed and if she has not voted yet, the clerk proceeds to the printing. The ID card is read and delivers an encryption of the voter's ID to the printer. The latter re-encrypts it (to avoid privacy attacks from a

[4] A difference to [26] is that we do not introduce separate Tracker Tellers, but instead let the Tally Tellers handle this, and we introduce a single separate Tracker Retrieval Authority TRA.

colluding ID card and scanner) and delivers a QR-code representing the ballot code $\mathsf{bcode} = (\{\mathsf{ID}_i\}_{\mathrm{OAEP},PK_T}, \{\mathsf{sign}_i\}_{\mathrm{OAEP},PK_T})$.[5] The signature is of the ID and the election ID, but can also include e.g. the date and the printer ID. The clerk should be screened from seeing the printed ballot, but can check that the correct ID card is read in the card reader.

After retrieving her ballot, the voter enters a booth and fills out the ballot with her vote $vote_i$ by hand.

The voter now proceeds to a ballot box that contains a scanner. The scanner first prints a receipt code, that is not yet detachable from the ballot box. This ensures that the receipt code does not depend on the vote and thus cannot be used as a subliminal channel. The receipt code is a random short pin, e.g. five digits, with check digits. The voter then puts her ballot in the box, the scanner reads it and releases the receipt code. It processes the data and re-encrypts the ballot code elements, encrypts the vote and receipt code, and publishes on BB (after election, if offline):

$$\{\mathsf{ID}_i\}'_{\mathrm{OAEP},PK_T}, \{\mathsf{sign}_i\}'_{\mathrm{OAEP},PK_T}, \{\mathsf{bcode}\}_{PK_T}, \{\mathrm{vote}_i\}_{PK_T}, \{RC_i\}_{PK_T}, \Pi_i$$

Here Π_i is a zero-knowledge proof of plaintext knowledge for the vote and receipt code and correct message space, for less malleability we suggest to include an AND-proof, proving that the two first encryptions are re-encryptions of the bcode in the third ciphertext. We include the election identifier in the hash of the Fiat-Shamir transform. The proofs will prevent vote copy attacks also across elections. The reason to re-encrypt the ciphertexts in the ballot code is to prevent coercion attacks via taking a picture of the filled-in ballot as a proof of the cast vote. In this case, a coerced voter can fill out a ballot as required by the coercer, photograph it, and go back to the officials for a new ballot and hand the (photographed) one, which is destroyed. They now cast their intended vote using the new ballot form. The re-encryption means that the paper ballot won't be linkable to the public electronic record, which is also important in the RLAs. Finally, $\{\mathsf{bcode}\}_{PK_T}$ is an encryption of the ballotcode which is here written in shorthand, but includes several ElGamal ciphertexts. If needed, these can be decrypted and allow crosschecking with the corresponding paper record.

4.3 Mix and decryption

These published tuples are now sent through a parallel mixnet (e.g. Verificatum [27]) on the BB after checking the proofs Π_i. After decryption of the first term we get back the ID and signature, i.e. we get mixed tuples of

$$\mathsf{ID}_i, \mathsf{sign}_i, \{\mathsf{bcode}\}'_{PK_T}, \{\mathrm{vote}_i\}'_{PK_T}, \{RC_i\}'_{PK_T}$$

Now the signature can be checked for eligibility verification and with the previous data on the BB we construct (suppressing re-encryption for clarity)

[5] Cryptographically it would suffice to leave out the encryption of ID_i, since it can be determined from testing different vk's.

$$\mathsf{ID}_i, \{g^{n_{\pi(i)}}\}_{PK_T}, C_i, \{\text{vote}_i\}_{PK_T}$$

As in Selene, the second and last term, the encrypted tracker and vote, are put through a verifiable parallel mix, after which the Tellers perform a verifiable decryption, to obtain the final tally board containing tracker/vote pairs:

$$(n_{\pi(i)}, \text{vote}_i)$$

4.4 Tracker notification

Receipt verification Before she can check her vote, the voter must enter the receipt code RC_i on her device (after log in). The app will encrypt the receipt code and a TT will do a PET between this encryption and the one displayed on the bulletin board. This verification is also done to ensure that the paper ballot and the corresponding electronic record are related to the same voter, i.e. to prevent an attack from the printer putting the wrong ID on the ballot, see section 5 for details.

Tracker retrieval The public commitment C_i and the corresponding α-term can be combined to form an encryption of the tracker under the voter's public key:

$$(\alpha_i, C_i) = (g^{r_i}, \mathsf{pk}_i^{r_i} \cdot g^{n_{\pi(i)}})$$

If the voter has entered the correct receipt code, the α_i term will be sent to the voter, and she can then compute the decryption using her secret key and retrieve $g^{n_{\pi(i)}}$, and hence her unique tracker $n_{\pi(i)}$. The Tracker Retrieval Authority will get the α_i shares from each Tally Teller (authenticated for accountability), multiply these together to obtain α_i and send this unauthenticated to the voter.

As described in Selene [26] it is computationally hard, without knowing sk_i, to calculate an alternative α-term that opens to a valid tracker. Thus the α-terms can be transmitted unauthenticated to the voter. On the other hand the voter can efficiently calculate such a fake α-term for any tracker (see [26]), and thus shows this to a coercer in case of coercion.

4.5 Risk Limiting Audits

A comparison Risk Limiting Audit (RLA) [19] is a method to confirm (or refute) the outcome of an election to any required confidence, by random sampling of the paper ballots. The digital and paper records of the vote are compared. Typically, for reasonably large margins, a small sample will suffice to achieve a good level of confidence, e.g. 95%. This technique requires a link between the digital and paper copies for every ballot.

The RLA testing, can be used to monitor the behaviour of the scanners. The audit should be performed in both directions, i.e. first start from tuples on BB, decrypt the bcode and find the corresponding paper ballot and check the consistency. In the other direction we can also start from a paper ballot, and the corresponding encrypted ballot can be found via PET tests, or more efficiently via an obfuscation of a part of the ballot code by lifting to a secret power homomorphically and then decrypting.

5 Preliminary Security Analysis

In this section, we give an overview of the security properties and their corresponding assumptions, and a brief, informal analysis of potential attacks. A more rigorous, formal analysis will be the topic of future work.

As is standard with E2E V schemes, we assume a trustworthy *BB* giving a consistent view for universal verifiability. This might be implemented as, for example [14], or perhaps using some form of Distributed Ledger Technology.

5.1 Verifiability

The current scheme has strong emphasis on verifiability and integrity, and like STAR-vote it has triple assurance by producing three lines of evidence for the result, firstly via tally board for the electronic ballots obtained via secure mixnets, secondly via the Selene mechanism that lets voters check their cast vote in the clear in the tally. Thirdly, the paper record enables comparison RLA checks between the paper and electronic records of the encrypted votes. Furthermore, the plaintext votes in the paper ballots can be hand counted if required. We also have universal eligibility verifiability via the ID-system, and the ID-marked paper ballots provide a solid base for dispute resolution.

The system remains secure even under large-scale collusions: even if the *BB* is malicious and can simulate views different views to voters, an honest RLA on paper ballots can detect a manipulated result. On the other hand, if all parties are corrupted including auditors, but the *BB* remains trustworthy and the Selene secret trapdoor keys are not compromised (e.g. stored on malware-free devices), then voters who check their vote can be sure it is counted correctly (assuming a computational assumption see [26]).

Universal Verifiability Universal verifiability allows voters and even third parties to verify the correct tallying of the votes on the *BB*. Anyone, including an interested voter, can check the proofs of the verifiable mixing and decryption of the encrypted votes and trackers. Further, the cryptographic operations on the *BB* during the setup phase of the Selene mechanism can to be verified by any observer. This will guarantee that each voter is assigned, and later notified, a unique tracker (assuming that the voter's Selene (trapdoor) keys are kept secret). As is true in general for e-voting schemes, the checks done by an individual voter are not enough to assure her that the overall outcome is correct. This is because she needs to be confident that all the other votes have also been correctly recorded and counted. This is assured if sufficient numbers of other voters have also checked their votes, or other mechanisms guarantee the correct handling of the votes. Here we have the possibility of auditors checking that the paper record is consistent with the digital record of encrypted votes on the BB. As an additional feature, as in STAR-vote, the interested party can also follow the evidence output by the RLAs that puts a risk-limit on the final result.

Eligibility Verifiability Eligibility is firstly checked by a registration clerk at the polling station, maintaining a log of attending voters. However additionally, the signature, produced by the smartcard and included in the ballot code, can be publicly verified on the bulletin board – even by third parties. Assuming secrecy of the signing keys, ideally protected with a pin on the smartcard, this proves the presence of the voter due to the existential unforgeability of the signature scheme, and prevents ballot stuffing. If the ID cards are used often, one concern might be that an adversary before the election maliciously obtains a signature of the data to be signed at the election. This can be prevented if the election ID contains some public random data first obtained briefly before the election.

Note that the tracker retrieval implicitly checks that the voter ID appeared on BB associated with some ballot, and then the check of the tracker on the tally board confirms correct vote. However the presence of their ID can also be checked independently by voters not using the Selene mechanism, or people abstaining. And anybody can check that only valid IDs and corresponding signatures appear giving the universal eligibility verifiability assuming an honest ID-card system.

Individual Verifiability Individual verifiability ensures that voters can check that their vote is recorded as intended. Typically, this is done by allowing a voter to challenge the encryption of the vote and then check the cast ciphertext on BB and then relying on the universal verifiability for the correct inclusion of the vote in the tally. The Selene mechanism used here, on the other hand, gives a more transparent direct verification directly of the cast vote in plaintext in the final tally. We hope that this will help incentivise more voters to do the check.

Attacks in the spirit of the trash attack [9] might appear troublesome. If the printer could see an ID associated to a voter, that is assumed not to perform any checks, it could then choose the ballot code (e.g. the last digits) to give instructions to a colluding scanner that it can maliciously change the electronic output vote. That is one reason, to let the ID card encrypt the ID, such that the printer won't be able to know the voter's identity. This attack thus requires a collusion between the ID card, printer and scanner and to identify a voter not doing the check. Further, the adversary also needs to collude with a tallier to miscount a vote in order to not get a discrepancy between the paper and online record. Further the attack could be caught by the RLAs. We see that the adversary needs a large collusion and has a risk of detection, whereas in normal paper voting the adversary only needs to control the tallier.

As with all E2E V schemes, the verification performed by an individual voter provides assurance that her vote is correctly handled but does not of itself provide assurance that the outcome of the election is correct. For that the voter needs to be confident that all other votes have been correctly handled. Usually this depends on a reasonable number of voters being sufficiently diligent in performing the appropriate checks, in particular checking that their encrypted vote appears correctly on the BB. This is perhaps a questionable assumption, but here, as with [20], we have an independent (paper) record of the cast encrypted votes which allows independent auditors to supplement the voter checks, lessening the dependence on voter diligence. This, along with the RLAs on the plaintext votes, should provide ample assurance to all that the outcome is correct.

Another attack vector is malware infection of the VSD for the Selene check, however it still requires a large collusion to actually change a vote.

5.2 Dispute resolution

One of the issues with the original (internet) Selene scheme was how to resolve a complaint when a voter claims that the vote they find on the BB is not the vote that they cast. If this occurs it is important for a judge to be able to determine where the complaint is genuine, arising from a corruption or malfunction of the system, or just a belligerent or forgetful voter. In Electryo, the election authorities can, in camera, request the corresponding ballot code on BB to be decrypted. The corresponding paper ballot can be identified using the ballot code allowing the vote, the ID and the corresponding signature can be checked.

5.3 Ballot privacy

Whereas we have high security guarantees for verifiability, the price for this has to be paid in terms of stronger assumptions for privacy. Ballot privacy essentially means that the voting system should not reveal a non-negligible amount of information about an honest voter's vote other than the outcome of the election. For a detailed overview of game-based definitions see [10], and also [24] for a recent work.

Clearly we need to assume that at least one Mix Teller is honest for ballot privacy, and that we do not have a threshold set of corrupted Tally Tellers. Also, if the ID card, the printer and the scanner collude, they can trivially map votes to IDs. However, if one of these are honest, then the adversary cannot link the IDs to the plaintext votes.

Further, each voter's supporting device is trusted for the privacy of the corresponding vote, but only if the voter actually uses the mechanism and enters the receipt code. This gives an extra layer of security for voters with high privacy concerns and who do not wish to trust the device and perform the check. It also safeguards voters with less technical knowledge, for whom the adversary has managed to setup and control the Selene keys and device. Note also that even if the secret Selene trapdoor key is leaked, the adversary still needs access to the α-term to break privacy.

The check of the additional receipt code via a PET is also used to guard against an attack where the adversary uses the Selene mechanism itself to check a vote, i.e. we must make sure that the Selene check is directed to the correct voter. Consider the case where the printer and smartcard are colluding, but the scanner is honest. To find the vote of the attacked voter, the printer uses the ID and signature key for a colluding voter and prints this on the ballot. Via the Selene mechanism this colluding voter can learn the vote of the attacked voter. This would be detectable via a Selene check done by the attacked voter, however the check of the receipt code means that this attack is detected with high probability and before the privacy leak happens, since the malicious voter does not know the correct RC_i.

The scanner itself can also carry out attacks on privacy by delivering a wrong output. Consider the case where the scanner sees an interesting vote, and wants to know the corresponding ID. It could then try to put the ciphertext of the ID in two different output tuples, and after the mixing and decryption of the IDs and signatures, it searches for two identical IDs. Likewise it could also change the ciphertext of the signature to produce an invalid signature, and then look for this on BB. Finally if the PETs of the receipt-codes are not done privately it could also provoke an error here, by encrypting a false RC_i. However, all of these attacks are immediately detectable with overwhelming probability and will deter a risk-adverse adversary from doing it. The RCCA security of the ciphertexts of the ID and signature means that it cannot do more advanced attacks than replacing or copying ciphertexts. The marking of ciphertexts used in [22] could also be introduced to make sure that the ciphertexts come from the ballot printer. Another worry could be that the scanner uses ciphertexts of votes from earlier elections, where they are publicly related to the IDs. However the ciphertexts in the output needs to come with proofs of plaintext knowledge, so the IND-CPA security prevents attacks of this kind.

Let us finally turn to the paper record. A problem here arises if a registration clerk manages to see the ballot code at registration, and is able to memorize part of it and to communicate this to a colluding tallier to break privacy. This is a reason to use QR-codes that are hard to memorise for humans, and we use printers with a shielding screen such that the clerk cannot see the printed ballot. In standard paper ballot voting, it is also not hard to make almost undetectable marks on the ballot. Alternatively we speculate that high resolution pin-hole cameras could be used by adversaries to photograph the ballots (backlit) before and after voting. The privacy could then be broken using that paper have a unique fiber structure (see e.g. [28]). We can never do better than the ordinary paper ballot scheme in this respect.

5.4 Receipt-Freeness and Coercion-Resistance

The receipt-freeness of a voting system says that the voting system should not give the voter any evidence to prove to a third party how she voted. The Selene e-voting protocol was carefully designed not to give such a receipt. However, it was only designed for use in low level coercion settings.

Firstly, the talliers of the paper ballots are trusted for receipt-freeness since the voter could make an almost invisible mark the ballot, a mark which the tallier will then look for. Likewise the scanner also needs to be trusted for receipt-freeness, since a malware could also be triggered with such a mark, something similar applies to the DRE machine of STAR-Vote, where a certain input pattern could alert the malware. However, the scanner could also do more direct attacks, it could e.g. store the mapping between receipt values and votes. This could be repaired by having a separate scanner of the QR code, and let this print the receipt code, and publish the encryptions on BB.

During the verification phase the TRA is trusted since it directly knows the true α-terms. The Tally Tellers however only need to be trusted to not be colluding in a threshold way which would break privacy.

We have also made sure that the ballot code only appears (re)encrypted on BB, and that the voters does not have a receipt showing the bcode.

The RLAs can also endanger receipt-freeness if not performed carefully, e.g. the ID should not be revealed in plain, but only the correct links via plaintext-equivalence tests. Also Italian attacks can be countered, see [8].

On the other hand, side channels attacks might occur. A coercer might be sufficiently convinced by a simple ballot selfie, and today little is done to actually guard against these. Alternatively it could also be printed after vote-casting on the filled-in ballot that have been folded to hide the voter's choice.

Finally note that a symbolic model of Selene has been proven receipt-free [12] for the case where we use auxiliary trackers per voter, see the longer version of this paper (to appear), where it is also shown how to request fake trackers before the end of the voting phase. Standard Selene suffers from a mild lack of coercion-resistance from the case where the coerced voter chooses a fake tracker which is the adversary's.

6 Conclusion

In this paper we propose a polling station voting protocol that combines a paper ballot record with a digital record to provide individual verifiability via the transparent Selene mechanism, universal verifiability and universal eligibility verifiability. Further the paper ballots are marked with the encrypted ID of the voters and provide a strong basis for dispute resolution and risk-limiting audits. However, as privacy and integrity are dual requirements, the strong integrity guarantees, especially for eligibility, comes at the cost of stronger assumptions for ballot privacy.

The hope is that the integrity guarantees and the transparent individual checks and good dispute resolution properties can be used to create trusted election results e.g. in cases where the integrity of earlier elections have been questioned.

For future work, the scheme certainly needs a full security model and corresponding analysis and proofs. Here it would also be natural to use automated verification in a simplified symbolic model, to make sure that attacks have not been overlooked. Cryptographically it would be interesting if the scheme could be improved such that we can move to a stronger version of the covert adversary model for privacy. Right now the adversary can pull off a privacy attack, but it will be detected. We could introduce a second authority which checks the decrypted IDs and signatures on BB before publishing these, such that the adversary would not be able to gain any knowledge from the attempted privacy attack. However, for integrity complicated zero-knowledge proofs are necessary when decryptions are not published, and efficient versions needs to be found.

Another important aspect for future research is the usability and user experience of the scheme, which should also include the officials.

Finally some variants of the scheme would be interesting to explore. Firstly, a postal version of the scheme. Postal voting is an increasingly popular voting form, but with integrity problems that should be addressed. Also, it would be interesting to explore a version of the scheme with everlasting privacy or participation privacy, but maybe milder guarantees for the universal part of the eligibility verifiability. Further, combining Prêt à Voter with the Selene check might have some advantages in privacy compared to the present scheme. Finally, the trusted platform problem is an important issue for the individual verification, and it would be interesting to explore the use of an extra device for the Selene check, e.g. utilizing the ID-card and a card reader.

Acknowledgements

We would like to thank the Luxembourg National Research Fund (FNR) for funding, in particular PBR was supported by the FNR INTER-Sequoia project which is joint with the ANR project SEQUOIA ANR-14-CE28-0030-01, and MLZ was supported by the INTER-SeVoTe project.

References

1. Estonia id card. `http://www.id.ee/`, accessed: 2017-11-10
2. An example of smart card implementing elgamal encryption with elliptic curves. `https://www.nxp.com/docs/en/data-sheet/P40C040_C072_SMX2_FAM_SDS.pdf`, accessed: 2018-05-11
3. German elections. `http://www.dw.com/en/german-election-volunteers-organize-the-voting-and-count-the-ballots/a-40562388`, accessed: 2017-11-10
4. Arnaud, M., Cortier, V., Wiedling, C.: Analysis of an electronic boardroom voting system. In: International Conference on E-Voting and Identity. pp. 109–126. Springer (2013)
5. Bell, S., Benaloh, J., Byrne, M.D., DeBeauvoir, D., Eakin, B., Fisher, G., Kortum, P., McBurnett, N., Montoya, J., Parker, M., et al.: Star-vote: A secure, transparent, auditable, and reliable voting system. USENIX Journal of Election Technology and Systems (JETS) **1**(1), 18–37 (2013)
6. Ben-Nun, J., Fahri, N., Llewellyn, M., Riva, B., Rosen, A., Ta-Shma, A., Wikström, D.: A new implementation of a dual (paper and cryptographic) voting system. In: Electronic Voting. pp. 315–329 (2012)
7. Benaloh, J.: Simple verifiable elections. EVT **6**, 5–5 (2006)
8. Benaloh, J., Jones, D.W., Lazarus, E., Lindeman, M., Stark, P.B.: SOBA: secrecy-preserving observable ballot-level audit. In: Shacham, H., Teague, V. (eds.) 2011 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections, EVT/WOTE '11, San Francisco, CA, USA, August 8-9, 2011. USENIX Association (2011), `https://www.usenix.org/conference/evtwote-11/soba-secrecy-preserving-observable-ballot-level-audit`

9. Benaloh, J., Lazarus, E.: The trash attack: An attack on verifiable voting systems and a simple mitigation (2016)
10. Bernhard, D., Cortier, V., Galindo, D., Pereira, O., Warinschi, B.: Sok: A comprehensive analysis of game-based ballot privacy definitions. In: Security and Privacy (SP), 2015 IEEE Symposium on. pp. 499–516. IEEE (2015)
11. Bernhard, D., Pereira, O., Warinschi, B.: How not to prove yourself: Pitfalls of the fiat-shamir heuristic and applications to helios. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 626–643. Springer (2012)
12. Bruni, A., Drewsen, E., Schürmann, C.: Towards a mechanized proof of selene receipt-freeness and vote-privacy. In: International Joint Conference on Electronic Voting. pp. 110–126. Springer (2017)
13. Culnane, C., Ryan, P.Y.A., Schneider, S., Teague, V.: vvote: a verifiable voting system (DRAFT). CoRR **abs/1404.6822** (2014), `http://arxiv.org/abs/1404.6822`
14. Culnane, C., Schneider, S.A.: A peered bulletin board for robust use in verifiable voting systems. 2014 IEEE 27th Computer Security Foundations Symposium pp. 169–183 (2014)
15. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Conference on the Theory and Application of Cryptographic Techniques. pp. 186–194. Springer (1986)
16. Goggin, S.N., Byrne, M.D., Gilbert, J.E.: Post-election auditing: Effects of procedure and ballot type on manual counting accuracy, efficiency, and auditor satisfaction and confidence (2012)
17. Jakobsson, M., Juels, A.: Mix and match: Secure function evaluation via ciphertexts. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 162–177. Springer (2000)
18. Küsters, R., Müller, J., Scapin, E., Truderung, T.: select: a lightweight verifiable remote voting system. In: Computer Security Foundations Symposium (CSF), 2016 IEEE 29th. pp. 341–354. IEEE (2016)
19. Lindeman, M., Stark, P.B.: A gentle introduction to risk-limiting audits. IEEE Security & Privacy **10**(5), 42–49 (2012). https://doi.org/10.1109/MSP.2012.56, `https://doi.org/10.1109/MSP.2012.56`
20. Lundin, D., Ryan, P.Y.A.: Human readable paper verification of prêt à voter. In: Jajodia, S., López, J. (eds.) Computer Security - ESORICS 2008, 13th European Symposium on Research in Computer Security, Málaga, Spain, October 6-8, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5283, pp. 379–395. Springer (2008). https://doi.org/10.1007/978-3-540-88313-5_25, `https://doi.org/10.1007/978-3-540-88313-5\_25`
21. Nemec, M., Sys, M., Svenda, P., Klinec, D., Matyas, V.: The return of coppersmith's attack: Practical factorization of widely used rsa moduli. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 1631–1648. ACM (2017)
22. Pereira, O., Rivest, R.L.: Marked mix-nets. In: International Conference on Financial Cryptography and Data Security. pp. 353–369. Springer (2017)
23. Phan, D.H., Pointcheval, D.: Oaep 3-round: A generic and secure asymmetric encryption padding. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 63–77. Springer (2004)
24. Quaglia, E.A., Smyth, B.: A short introduction to secrecy and verifiability for elections. CoRR **abs/1702.03168** (2017), `http://arxiv.org/abs/1702.03168`

25. Ryan, P.Y., Bismark, D., Heather, J., Schneider, S., Xia, Z.: Prêt à voter: a voter-verifiable voting system. IEEE transactions on information forensics and security **4**(4), 662–673 (2009)
26. Ryan, P.Y., Rønne, P.B., Iovino, V.: Selene: Voting with transparent verifiability and coercion-mitigation. In: International Conference on Financial Cryptography and Data Security. pp. 176–192. Springer (2016)
27. Wikström, D.: User manual for the verificatum mix-net (2013)
28. Wong, C.W., Wu, M.: Counterfeit detection using paper puf and mobile cameras. In: Information Forensics and Security (WIFS), 2015 IEEE International Workshop on. pp. 1–6. IEEE (2015)

Lessons Learned

Faults in Norwegian internet voting

Christian Bull[1 [0000−0003−2363−9918]], Kristian Gjøsteen[2⋆], and Henrik Nore[1 [0000−0002−3187−1664]]

[1] New Voting Technology Consulting AS
{christian,henrik}@nvtc.no
[2] Norwegian University of Science and Technology
kristian.gjosteen@ntnu.no

Abstract. Norway ran large-scale trials of internet voting during the 2011 local elections and the 2013 parliamentary elections. In 2016 and 2018, several municipalities and one county have used internet voting for local referendums. There were a number of faults, from misprinted voting cards via faulty pseudo-random generators to misconfigured certificates. We discuss the impact these faults have had directly and indirectly, in terms of lost votes and negative public relations, among other things.

1 Introduction

Voters using their own internet-connected terminals to vote in political elections or referendums is often called *internet voting* or *remote voting in uncontrolled environments* [2]. Compared to many other computer systems, a system for internet voting is fairly simple. However, the context of political voting means that faults can much more severe consequences than for most other systems of comparable complexity. Understanding faults and their consequences is therefore important, both for practitioners and theoreticians.

Norway ran large-scale trials of internet voting, first during the 2011 local elections [1, 3] and then during the 2013 parliamentary elections. These elections were run by the central government on behalf of the municipalities (who run elections in Norway), using software from an international e-voting vendor.

During 2016 there were a number of local referendums in municipalities, many of which made use of internet voting. And Finnmark fylke (county) held a referendum in May 2018, also making use of internet voting. These referendums were all run using software from international e-voting vendors.

So far, every Norwegian political election and referendum using internet voting have also had the option of traditional paper voting. This has both been for accessibility reasons (not everyone has internet access, or want to use internet voting) and as an anti-coercion technique (voters are allowed to vote only once on paper, but as many times as they like on the internet; if a voter has submitted a paper ballot, that paper ballot is counted, otherwise the last submitted internet ballot is counted).

⋆ Funded by the Norwegian Research Council, project 275516, *Secure, Usable and Robust Cryptographic Voting Systems*

The trials in 2011 and 2013 explicitly did not attempt to increase turnout, and there is no evidence to suggest that voter turnout did increase. However, the later local referendums increasing turnout was the main reason to use internet voting, and evidence suggests that internet voting did increase turnout significantly. This is consistent with experience from other countries.

2 Faults

One municipality has used internet voting in 2011, 2013, 2016 and 2018, and the use of internet voting is rising. While some faults did not affect this municipality, it suggests that these faults had little impact on public confidence in internet voting.

Late start. Both in Ålesund in 2011 and in Finnmark in 2018 the start of internet voting was delayed. These delays were not very significant (30 minutes in 2018).

In 2011 it seemed as if nobody noticed the delay. This is hardly surprising, since internet voting was very new in 2011 and most voters were either not in a hurry to cast their ballot, or accepted that new services often have teething problems. The 2018 delay was noticed and resulted in negative reports in the local newspapers, but nothing very significant.

Misprints. The 2011 and 2013 elections relied on so-called return codes to detect compromised terminals. These were printed on the voters' poll cards. Misprints on these cards in 2011 led to many voters getting incorrect or unreadable return codes printed on their poll cards.

Norwegian news organisations did not seem to care about these misprints. However, on the plus side, the fault essentially turned the election into a natural experiment on whether voters actually would report non-matching return codes. The number of voters who called the helpdesk due to non-matching return codes is known, but the exact number of misprinted poll cards is unknown. Also, we can only estimate how many voters used internet voting and had misprinted poll cards. However, the efficiency estimate this gives us suggests that about 85% of voters were able to use the return codes correctly, a surprisingly high number.

Cryptographic errors. In 2013 a faulty pseudo-random generator caused a total failure of the cryptography protecting ballot confidentiality [2].

Fortunately, there were multiple layers of security in the system, so the practical consequences seem to be nil. Curiously, the Norwegian press completely ignored this cryptographic failure, suggesting that it did not in any way affect voters' confidence in the election organisers.

Infrastructure problems. For some of the 2016 referendums, there were significant stability issues with the software used, essentially requiring a restart of the systems every few minutes.

These restarts were noticable by voters and resulted in significant negative attention from local newspapers. However, the referendum organisers (the municipalities) seem to be satisfied with the overall effect of internet voting.

Configuration errors. During the 2018 referendum, some voters could not vote because of a certificate configuration error. The error was quickly corrected.

This case is special, since the logs actually tell us which voters encountered this problem (about 3%). The logs also tell us which of them later managed to vote via the internet (about 2%) or on paper (about 0.2%).

We can compare the loss of 0.8% of votes to the abandonment rate for voters who did not encounter this configuration error (about 0.6%), which means that the configuration error probably doubled the effective abandonment rate.

The error had no effect on the outcome (yes vs. no), but it may have had a small effect on the exact number of votes for each side.

Even though everything was disclosed and a fairly large number of voters were involved, the public seems not to have noticed.

3 Discussion

Voter abandonment The 2018 configuration error gave us some information about voter abandonment because of errors, and it seems to be greater than the general abandonment rate. Again, this error turned into a natural experiment, which gives us some information about how easily voters can be turned away.

This emphasizes the need to ensure quality control. Especially system developers must build systems that are easier to configure correctly, since testing cannot be expected to uncover every configuration error.

Implementation faults The faulty pseudo-random number generator from 2013 emphasizes the need for vendors to adopt good methodologies for secure development. Also, the need for sufficient time for development is important, which is an important lesson for adopters, since it is usually undesirable to postpone elections. Finally, multiple layers of security has yet again proven to be useful.

Public perception Even though the voters seemed annoyed by the infrastruture problems from 2016, and also somewhat annoyed by the late start in 2018, there seems to be little or no damage to the confidence in internet voting or in the willingness to use internet voting. The most likely explanation is that this was an availability issue, and voters in general have a lot of experience with the non-availability of lower-quality public web sites and other infrastructures. Availability issues as such will therefore not be a surprise, even though they are annoying.

The misprints in 2011 had significantly more potential to produce negative publicity, but there were two issues. First, certain system properties guaranteed that this could not be the symptom of a straight-forward attack. (While in theory such misprints could be used to disable the return code mechanism for a given voter, such an attack would be quite difficult to execute.) The policy of transparency certainly helped explain the situation to the affected voters, namely that the incorrect return codes did not prove that an attack was involved.

Second, the public discussion (such as it was) had been dominated by the issue of coercion. The misprints were not at all relevant for coercion, so there was no media interest.

As far as the organisers were concerned, these events did not affect the outcome, and even though bugs probably caused some voter abandonment, it was completely negligible compared to the increase in turnout internet voting gave.

4 Concluding Remarks

In internet voting, practitioners need a realistic appreciation of the potential faults and their consequences, even when using systems from established vendors. Theoreticians need to be aware of these issues with deployed internet voting systems, in order to design more robust systems. The experience with internet voting in Norway is therefore useful both for practitioners and theoreticians.

References

1. Kristian Gjøsteen. The Norwegian internet voting protocol. In Aggelos Kiayias and Helger Lipmaa, editors, *E-Voting and Identity - Third International Conference, VoteID 2011, Tallinn, Estonia, September 28-30, 2011, Revised Selected Papers*, volume 7187 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2011.
2. Feng Hao and Peter Y. A. Ryan, editors. *Real-World Electronic Voting: Design, Analysis and Deployment.* CRC Series: Series in Security, Privacy and Trust. CRC Press, 2016.
3. Oliver Spycher, Melanie Volkamer, and Reto E. Koenig. Transparency and technical measures to establish trust in Norwegian internet voting. In Aggelos Kiayias and Helger Lipmaa, editors, *E-Voting and Identity - Third International Conference, VoteID 2011, Tallinn, Estonia, September 28-30, 2011, Revised Selected Papers*, volume 7187 of *Lecture Notes in Computer Science*, pages 19–35. Springer, 2011.

Elections and IT; the challenge of making it work in a changed world

Peter Castenmiller[1] and Pamela Young[1]

[1] Dutch Electoral Council P.O. Box 20011, 2500 EA Den Haag, Netherlands,
p.castenmiller@planet.nl

Abstract. In this chapter, the reader is informed about the latest developments in securing the integrity of the election outcomes in the Netherlands, and especially about the uses of the software that supports the election process. It is argued that there should be more international cooperation in safeguarding the integrity of the election process.

Keywords: Elections, Politics, Information technology

1 Introduction

Most people associate the use of digital means in the election process with voting devices or internet voting. Yet there is so much more, but this gets little or no attention. During the last two years the use of software to support the voting process became, quite unexpected, a major issue in the Netherlands. In the case of the national elections, in the spring of 2017, this almost led to a major crisis [1]. In 2018, there were municipal elections, and once again, there were problems. In this article, we first will inform the readers about the new challenges that emerged during these local elections.

As there are discussions about the use of software in the election process for two years in a row now, this suggests that we are dealing with a structural problem. Our second purpose in this article is to describe the underlying problems and to stimulate the, in our opinion much neglected, attention for the use of software which supports the paper-based approach in the election process.

2 The Dutch case in general

Countries differ in the ways they arrive at election results. In the first place, this is the consequence of differing electoral systems, of course. There are also differences arising from other structures and from the collaboration of bodies charged with the responsibility for the result. There are also differences among the logistical processes that establish the results. One important element of those logistical processes is the degree to which digital resources are used to establish the result. As most countries still count the votes manually, many, at least partly, also rely on digital processes and the use of software to aggregate votes.

In the Netherlands, the election process has had considerable public interest in the past years. Some fifteen years ago, it appeared that the use of voting machines would become the standard. After the year 2000 more and more municipalities started to use electronic voting devices. Yet, after 2006 it was established that using these devices resulted in an insufficient level of transparency regarding the outcome of the elections. They were therefore abolished [2].

So, in the Netherlands, all votes are cast on paper in elections by coloring the preferential candidate's box red. Immediately after the polling stations' closing time, at 9 PM, all paper ballots are counted manually at the polling station itself. The result is recorded in an official report. The chairperson delivers this official paper report in person from the polling station to the municipality that very same night. All the official reports are collected and counted at the town hall – often deep into the night – so that the result of an election is known at the municipal level.

When it comes to national elections, the municipal results are delivered on paper to the Electoral Council after an administrative process that takes a few days; the Electoral Council then establishes the results. It has been known for many years that the logistical challenges associated with this paper process demands a considerable amount of work from those involved. As mentioned, it takes a lot of time. Furthermore, the process in which results are handed over on documents filled in by hand can lead to several errors. It is clear that the paper process and the manual count have vulnerabilities. In the national elections of 2017 there occurred an incident that led to the loss of some ten thousand votes. This incident was described in a previous contribution for the Bregenz Conference in 2017 [3]. So, despite the best intentions and extensive instruction, errors are made with the manual counting and the subsequent transfer of results into official reports

In the Netherlands, many of the logistical challenges associated with the elections are the responsibility of the municipalities in the first instance. Since the reintroduction (after 2006) of the traditional paper-based approach of voting, most municipal employees have stayed strong advocates of digitization of the election process. After struggling with the ‘paper process’ for more than ten years after the abolishment of voting devices several interest groups from municipalities sounded the alarm and drew up an Election Agenda with a list of priorities that they feel must be implemented by 2021. One of the desires high on this agenda is electronic ballot-counting. There is also a plea for electronic voting. In this, municipal representatives emphasize primarily the need for speed and a more efficient process. Various interest groups of municipal representatives emphasize that technology presents opportunities for a faster and easier voting and results tabulation. They also expect that the use of technology can lead to a higher accuracy, which could enhance integrity and boost public trust in the outcome. In a general sense, the claim is made by these interest groups that the electoral process must "stay up-to-date".

The main use of this Election Agenda is to stimulate the discussion about reforms in the election process. Yet, in these claims, little attention is paid to the vulnerabilities that digital resources also introduce. The context in which elections are organized and the role of digitization in this have changed considerably during the last years. The organization and conduction of elections are receiving increasing public scrutiny - in the Netherlands in any case. Faith in an orderly course of the elections and in correct results is no longer self-evident, as was the case before 2017. Various possible causes can be considered. To start with, decreased confidence is fed by the worldwide discussion about "external powers" affecting elections. By extension, there is also attention to fake news and concealed influence via such social media as Facebook. More generally, there is greater awareness of cyber-security threats and the associated vulnerabilities of hardware and software used in the chain. This goes for several countries, notably the American presidential elections of 2016, but also for the Netherlands.

In the Netherlands, there is another development that leads to more attention for the way in which the results of elections are established. Increasing political divisions also put the process under pressure. For example, "losers" may benefit politically from questioning the reliability of the elections and the result. Social media offer an easily accessible and extremely effective platform for this. In the process of elections, in which confidence is a large component, it ultimately makes little difference whether the undesirable influence is a true danger or simply a perception.

In the Netherlands, over the course of the years, several measures have been taken to manage the current challenges to the electoral process. One important instrument in this range is the Electoral Support software (Dutch: Ondersteunende Software Verkiezingen - OSV), developed in 2008,[4] after the use of electronic voting devices were forbidden and first used in the European elections of 2009. The Electoral Council is the administrator of this software. In practice, nearly all political parties and municipalities use the software, although this happens on a voluntary basis

The software is first used to support the nomination process. The registration of all participating candidates forms the foundation for all the other steps and official reports that are included in the process. Political parties that want to participate in the elections are offered the possibility to use the software to register their candidates. The software is set up so that, on the basis of the inputted political parties and candidates, various models for the official reports can be created and printed throughout the various stages of the process.

Furthermore, the software is used for the vote tabulation and seat distribution. It is important to point out that this software is not used at the level of the polling stations. As mentioned, the votes cast on paper are first counted manually. It is only once the results are in the town halls that software plays a role. The results from each polling station are manually entered twice into the software which is installed on computers that may not be connected to internet. The software determines the results for the municipalities and prints them on paper. This process is repeated at the district level by the principal electoral committees and eventually by the Electoral Council.

At various moments during the process, results are printed on paper, are brought to the next level in person and manually re-entered into the system. Up until the 2017 election it was also standard procedure that a digital file of the results was transferred together with the paper print by using usb-sticks. This is not done anymore as an extra safety measurement.

Although elections in the Netherlands are still paper-based, this software nowadays plays a key role in the election process. We assume that similar software will be used in other countries. Yet at the Dutch Electoral Council we hardly have any knowledge of the use and experiences in other countries.

3 The experiences in the elections of 2018

Previously, during the 2017 Dutch parliamentary elections, the process of establishing the outcome unexpectedly took center stage. Shortly before the elections, a national news program brought attention to the presumed unreliability of the software to be used in the elections. Fundamental questions were posed about the reliability of the results and the role of the supporting software used for this. The news item had a significant effect and resulted in parliamentary discussions. This ultimately had consequences for the software's method of use as well. The events, and especially the improvisations that resulted from these in order to establish an official and reliable result within the deadline nonetheless, were described extensively in a paper for the 2017 E-ID conference [5]. As was mentioned in this paper, the situation was complex and unexpected. A month before the elections it wasn’t clear which way the results should be determined. At the last minute, some new procedures were required to safeguard the processes and results. The new procedures insisted on a minimum use of digital means.

Dutch elections were held once again in the spring of 2018, this time for the municipal councils. The municipalities themselves were responsible for those elections. A vote was also held at the same time for a corrective advisory referendum. These were elections for which the Electoral Council carried final responsibility. In the run-up to the 2018 municipal council elections and to the referendum, this same news station paid attention to the reliability of the software. For this news item, this news channel had asked a number of "ethical hackers" to evaluate the source code of the software used. In the opinion of this news channel, this resulted in the evidence of some 50 shortcomings.

Nonetheless, after all the commotion in 2017, things had indeed already been changed. To start with, the Electoral Council had already had the software intensively evaluated in 2017 by Fox-IT, a company specialized in auditing and securing software, and by SQS, a company specialized in software quality. This had already resulted in the implementation of improvements at that point.

But certainly not all of the problems had been resolved. After all, the structure of the software and the technologies used dated from 2008. In the course of a decade, insights into the software and into what makes software reliable have changed, along with the context of its use. It would require a major operation to adapt the election software to this. The Electoral Council, the administrator of the software, is aware of this and believes that new software should be developed in the short term on the basis of more stringent regulations. Yet, according to the Electoral Council, the risks were acceptable, provided that everyone involved in the use of the software would stick to the procedures. In the end the results of the elections were determined without any major problems [6].

4 Integral approach to the risks

Will the development of new software alone be sufficient to make the process sufficiently robust again? Some of the vulnerabilities relate not to the software itself, for example, but to the hardware[7] that the municipalities use to structure the associated processes. This makes it clear that modernizing only the software in order to establish the outcome is not sufficient. Monitoring the electoral process integrally for risk is permanently needed, along with monitoring and mitigating digital threats wherever possible, in order to make the process future-proof and to keep it that way.

In order to make the process future-proof, the existing regulations must also be addressed. The last integral revision of the Dutch Electoral Act dates from 1989. The principles formulated in that regulation no longer fit with the drastically changed and now far more complex context in which elections are held these days. Furthermore, in 1989, no one could foresee the nowadays widespread use of hard- and software. Nearly all current regulations presume a process in which digitization plays no role. In actuality, the changed context makes it virtually impossible to establish results within the deadlines without the use of software. With this in mind, it is curious at the very least that this is not acknowledged in the regulations. So it would be naïve not to include a feasible plan in the regulations to fall back on if this is considered necessary due to external threats.

Finally, it also demands a different view of the speed with which definitive results come about and are publicized. Even if it may be normal in many countries to have a final result be established only after a few weeks, in the Netherlands, the results are irrevocably established within a week after a vote and the new members of Parliament are installed within eight days after the election. So it will be clear that this allows for little or no latitude for serious research into errors in the process, let alone investigating secret undesirable influence for fraud.

5 Whose turn is it?

While unlimited faith in the possibilities and results of digitization may seem to have existed 15 years ago, that picture has been considerably altered in recent years. More attention is now paid to the less positive sides of large-scale digitization such as hacking and other forms of influence. This implies that more attention must be paid to the quality and reliability of digital support and resources. With this, the development of new, reliable digital resources (both hardware and software) has become a complex assignment in which many players must be involved. In addition to government (municipal and federal levels), this also concerns such new players as security services, CIOs, cyber-security experts and (even) ethical hackers.

The procedures surrounding elections must be structured in such a way as to recognize risks prior to an election and to mitigate these wherever possible. In addition, mechanisms must be present to adequately correct errors afterward and to detect fraud. The EMBs responsible for establishing the result have a role in placing the risks on the agenda. These parties cannot be expected to foresee all of the risks themselves, simply because knowledge about the current threats is missing. Yet they are ultimately responsible for the conduct of the elections. Therefore, they need to work with other institutions to ensure all aspects are sufficiently covered.

As is often the case, government has an important role to play. The decision about whether elections should be viewed as part of a country's vital infrastructure, for example (as is nowadays the case in the United States), requires political consideration. And the balance between the vulnerabilities of paper and of digital processes has a particularly political character. This goes for the consideration of which process (paper or digital) should ultimately be decisive in the event of differences, for example. Finally, residual risks that are difficult to mitigate must be acknowledged by the government and be discussed and - perhaps even more important - also be accepted.

All of this demands a different and broader governance model from the one that we have become used to date. The European Commission for democracy through law (Venice Commission), the International institute for democracy and electoral assistance (International IDEA) as the International foundation for electoral systems (IFES) have all recently addressed the need for a broad, multi-disciplinary collaboration when using technology in the electoral process. This is needed to further strengthen the security of election technologies, and to become more resilient against emerging cyberthreats.[8] In processes with so many players and often turbulent conditions it is even more important to lodge separate responsibilities within the multi-disciplinary approach with parties who can (continue to) honor that responsibility in practice. This means that responsibilities, particularly in the digital process, must be assigned and delineated more than is the case at this moment. The various specialist domains and knowledge currently present in government can then be better used. Consider, for example, security services and departments that already concern themselves with cyber-security daily. That knowledge is indispensable. Precisely now, with the structuring and structural maintenance of a robust election process and not simply after a crisis has occurred.

Since the authority concerning elections is allocated among various parties in the Netherlands, it is not immediately evident who is responsible for the development and maintenance of digital resources such as the software used to determine the result. Experience has shown that the responsibility for large ICT projects is a difficult one. The substantive challenges and complications are huge. Although municipalities, the ministry involved and the Electoral Council itself acknowledge and support the need for modernization and better support of the election process, immediate differences in insight then arise about the approach and the responsibilities of the various parties involved.

It can be argued that the guidance and coordination of digital resources in the electoral process should be lodged within a body that oversees the responsibility not only for the entire system, but that can also actually substantiate this. In the Dutch situation, that should be the Minister of Internal Affairs. That person carries political responsibility and has the required multidisciplinary and government-wide resources to actually shoulder that responsibility. At the same time, when bearing such a responsibility, immediate tension arises with the importance of safeguarding the required independence with the actual establishment of an election result. Because of that tension, the Dutch Minister currently elects not to adopt a predominant role.

It is clear however, to everyone in the Netherlands who is involved in the election process, that things will have to change. At present, the discussion revolves around the question who is responsible or should take the lead. But after this will be established (which in itself might still take a year or so), new software has to be developed. And new digital means and procedures will be introduced in the election process, although it is not quite clear which ones these will be. However, the use of electronic voting devices is not expected.

6 From 'the worst' to 'the best' of both worlds?

There are good reasons to stick with the paper process in the Netherlands. Auditability and transparency are now precisely the significant advantages to the emphasis on paper. The process can then always be reconstructed, and it is much more difficult to affect the results systematically. But that's not to say that it won't represent a lot of work and a huge amount of time.

At the same time, the digital process has also proven its added value in determining the results. Considerably more yield can nonetheless be gained from a modernized version of the software for establishing the result. Software is excellent for use in filing off the sharp edges of logistical challenges and in preventing errors and mistakes. One possibility is an audit of the entry (are implausible values being entered, for example), or providing transparency by making the digital and scanned paper files of the results available. If the results of intermediate steps for arriving at the final results are regularly published on Internet using software, then everyone can also check each step. This makes systematic influence much more difficult.

We assume that other countries are also dealing with the challenge of organizing reliable elections in the present-day context and are also wondering to what extent they can make use of software and computers. That's why we are also interested in hearing whether these challenges are also familiar in other countries. In the long run, we would like to share the various experiences. On that basis, a 'Code of best practices' might be established.

After all, organizing and maintaining reliable elections are too important to let individual countries deal with the various challenges all on their own.

References

1. Castenmiller P. and Uijl K., The use of supporting software in elections, the peculiar case of the Netherlands 2017. In: R. Krimmer et al (eds.), Proceedings E-Vote 2017, pages 315-325. https://www.e-vote-id.org/wp-content/uploads/2017/10/TUTPress-2017.pdf
2. Loeber L., E-Voting in the Netherlands; from General Acceptance to General Doubt in Two Years, Electronic Voting, 2008.

3. http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=D2F428B5F4018C50E4C26E73 93A8EEFF?doi=10.1.1.464.9460&rep=rep1&type=pdf
4. Castenmiller P. and Uijl K., The use of supporting software in elections, the peculiar case of the Netherlands 2017. In: R. Krimmer et al (eds.), Proceedings E-Vote 2017, pages 315-325. https://www.e-vote-id.org/wp-content/uploads/2017/10/TUTPress-2017.pdf
5. The development was initiated by the Dutch Electoral Council, with the support of the Ministry of Interior Affairs. A software company, that has also developed election software for Germany, did the work.
6. Castenmiller P. and Uijl K., The use of supporting software in elections, the peculiar case of the Netherlands 2017. In: R. Krimmer et al (eds.), Proceedings E-Vote 2017, pages 315-325. https://www.e-vote-id.org/wp-content/uploads/2017/10/TUTPress-2017.pdf
7. Yet, in several municipalities there appeared some ‘close calls’. Sometimes, a difference of less than ten votes had an important political impact. This provoked a widespread call for recounts. In more than twenty municipalities, these recounts took place. It is important to stress that none of these recounts took place because there were doubts about the software.
8. The municipalities are expected to use computers that have no direct connection with the internet. Furthermore, they have to use the latest versions of anti-virus software.
9. More information can be found on the following websites:
   - Venice Commission: http://www.venice.coe.int/webforms/documents/?pdf=CDL-EL(2018)001syn-e#
   - IDEA:https://www.idea.int/news-media/news/cybersecurity-and-elections-international-idea-round-table-summary
   - IFES:http://www.ifes.org/sites/default/files/ifes_erben_raising_trust_in_electoral_technology_innovation_aided_by_traditional_approaches_d7_apr_2018.pdf

Evaluating Practical Experiences

The E-voting Readiness Index and the Netherlands

Leontine Loeber

University of East Anglia
Leontine_loeber@xs4all.nl

Abstract. In this paper the four dimensions of the E-voting Readiness Index are applied to the Netherlands. It examines how the Dutch systems should be scored when it comes to political willingness to introduce “e-voting”, the legal system concerning elections, the existing technological level and the societal aspects concerning “e-voting”. Special attention is given to the trust that voters stated to have in different voting technologies during the Dutch Parliamentary Election Study held during the 2017 Parliamentary Elections. In conclusion, even though the Netherlands scores relatively high on the technological dimension, the current state of the political, legal and societal dimensions do not point towards a likely adaptation of “e-voting” in the near future of the Netherlands.

Keywords: E-voting Readiness Index, case study, trust, “e-voting”, the Netherlands.

1 Introduction

Although a majority of countries that hold elections seem to be using forms of technology within their election process, the use of technology to actually cast a vote is relatively low [1]. Casting a vote through technology in this context means voting through electronic means in a polling station or through the internet.[1] Even though a number of countries have considered introducing “e-voting”, not many have done so, where as some countries have even stopped using “e-voting” [2, 3, 4, 5]. So what determines if a country is likely to introduce “e-voting”? To examine this, Prosser and Krimmer introduced criteria to assess and compare different “e-voting” initiatives [6]. This model was further developed into the E-voting Readiness Index by Krimmer and Schuster [7]. Although the model is very useful [8], this has not lead to a substantial body of literature in which it is applied to different countries [9, 10]. To add to the existing literature, in this paper the index is applied to the Netherlands. This paper could then be used to compare this country to other countries for which the index is applied.

[1] In the literature sometimes the first form is described as “e-voting” and the second as “i-voting”. In this paper the two forms together will be referred to as “e-voting”.

2 The E-voting Readiness Index

As described by Prosser and Krimmer, the model focuses on all relevant areas to determine if a country is ready and likely to introduce “e-voting”. The model differentiates between four separate dimensions: (i) Politics, (ii) Law, (iii) Technology, and (iv) Society. The model distinguishes between a project and a national level to prevent pilot experiences to be mistaken for national experiences [6].

For the Politics dimension, the kind of political system (constitutional monarchy, parliamentary democracy, etc.) should be taken into account, the method and frequency of elections as well as general statistics on elections (number of eligible voters, electoral districts, polling stations etc.). Another important aspect of this dimension is the attitude of the government and parliament toward “e-voting” and more in general toward “e-government”. Other factors that are included in this dimension are the current stage in the policy making process with regards to “e-voting”, the aim of the policy if in place and if an official organisation is planned for the implementation of “e-voting” [6].

Key element for the Law dimension is the kind of legal system that exists within a country. Most relevant is whether the electoral law provides a basis for technological solutions within the election process. This means that the existing legal principles for elections are important, possible ways “e-voting” could be implemented in the legal framework and the stage in which “e-voting” is in the current legislation-making process [6].

The third dimension, Technology looks at the status of registers in general and the register of citizens and of eligible voters in particular. Other important technological infrastructure questions are if a country has implemented a digital national ID card, digital signatures and if international “e-voting” standards are or will be adopted. This dimension also looks at the level of “e-government” services in general [6].

The last dimension looks at society. This dimension focuses on the level of political participation, the turnout for voting and the attitude of citizens towards new technologies and “e-voting” in particular. To make an assessment of this dimension, it is also important to know the penetration rate of mobile phones, personal computers and tablets and the Internet. An interesting factor is the actual use of Internet in society [6].

The model shows similarities with models developed to test the capacity of countries to adopt “e-government” [11]. In these models, key indicators are the country’s political will, the availability and strength of their human capital, the ICT (telecommunications) infrastructure, and the presence of administrative priorities. The UN used these factors to present an “e-government” index, that that reflects the ‘requisite conditions’ that contribute to establishing an enabling environment for “e-government” [12].

3 Data

The data used to measure the societal dimension (described in paragraph 8) stems from the Dutch Parliamentary Election Survey 2017 (known in Dutch as the NKO 2017). The Dutch Parliamentary Election Study is conducted in two waves before and after elections for Parliament interviewing the same group of persons. For each Study, persons are randomly selected out of the all the voters that are eligible to vote in those Parliamentary elections. Selected persons are interviewed in person twice, a few weeks before the election and directly after the elections. These interviews are conducted at the homes of the participants. After the second interview, the participant is also asked to fill in a short written questionnaire. Also, for some questions drop-off forms are used. The Dutch Parliamentary Study contains nearly 700 questions on a wide range of subjects. During each study certain questions are added or removed, based on current events [13].

4 History of “e-voting” in the Netherlands

One of the reasons why it is interesting to apply the E-voting readiness index to the Dutch case is the history the Netherlands has had with “e-voting”. The country introduced voting computers (DRE’s) in the early 1960’s and continued to use these until 2006. It also experimented with internet voting for voters living abroad and in 2005 the general wish of parliament was to introduce internet voting for all Dutch voters. In 2006 however, an action group successfully challenged the certification of the voting computers, claiming that they were not meeting the standards of transparency, verifiability and voter secrecy. The main problem that the action group had with the machines in use was the fact that they were “paperless”, meaning that there was no way to check if the outcome of the election was indeed what the voters wanted. There is however no international legal obligation for governments to provide integer elections. They do have to ensure secrecy of the vote. Therefore, the action group also showed that it would be possible to breach the secrecy of the vote since the radiation (or Tempest) of the screens of the voting computers could be “read” with a handheld device. They thus challenged the use of the computers in court, leading to a withdrawal of the certification and a return to voting with paper ballots [2, 3]. At the same time, internet voting was considered for nationwide elections for the waterboards, a form of decentralized governments. Because of the discussion on the voting computers, a more substantial technical analysis of the intended system was performed, showing several weaknesses. This led to the decision not to use the internet voting system anymore [2, 14]. Since that time, several attempts by the central government, members of parliament and municipalities have been made to re-introduce forms of “e-voting”, but so far none have been successful [14]. Given this history, application of the index could perhaps shed some light on the question why there has been no success yet and if the country is ready for re-introduction of “e-voting”.

5 Political dimension

5.1 Political system

The Netherlands is a constitutional monarchy. Legislative power is held by a bicameral Parliament. The First House (Eerste Kamer or Senate) consists of 75 members, who are elected for a four year term by the 12 Provincial Councils. The Second House (Tweede Kamer or House of Representatives) consists of 150 members that are elected every four years on the basis of a proportional system. The Second House has greater legislative powers than the First House, but the First House still has veto power over every bill. The Head of State is the Monarch, whose function is largely ceremonial. Executive power is exercised by the government. Based on the election results of the elections for the Second House, a coalition is negotiated. The Prime Minister is usually the party leader of the biggest party that is a member of the coalition. Part of the coalition agreement is the division of posts in the Council of Ministers or the Cabinet. The Cabinet plans and implements the government's policy. The Ministers are responsible to the Parliament, both collectively and individually. If a Minister loses the trust of Parliament, he or she will usually resign and a new Minister is appointed. The government however can also choose to disband the Second House and call for new elections.

The local Government in the Netherlands consists of 12 provinces and around 400 municipalities. They are governed by a locally elected provincial or municipal council. Elections for these councils are held every four years, in different years for provinces and municipalities. Dutch citizens also elect the Dutch members of the European Parliament, every 5 years.

5.2 Politics and "e-voting"

The political dimension of "e-voting" in the Netherlands can be characterized as one of extremes. As stated before, in 2005 practically the whole Parliament was in favour of introducing internet voting for all Dutch voters. After the problems with the voting computers in 2006, the political will to use technology in the voting process seems to have disappeared overnight. However, this didn't last long as soon after certain parties with Parliament already expressed to be in favour of re-introducing voting computers and even internet voting [14]. In the coalition agreement of the last government, specific mention was made of the use of technology in the voting process. Although a lot of debates have been held since 2006 and now on this topic in Parliament, there has not been a majority that managed to introduce new forms of technology in elections. The latest government has not mentioned "e-voting" in the coalition agreement.

However, when it comes to "e-voting" it is not just the national political dimension that should be considered. In the Netherlands, the municipalities are very involved when it comes to organizing elections, since they are given their own tasks in the Election Law. Because the municipalities are the bodies that actually have to find people that are willing to be poll workers and are also responsible for ensuring that voters can go to suitable polling stations that are also accessible for voters with a handicap, the most pressure to re-introduce “e-voting” seems to come from politicians and majors on the local level.

Interestingly, it is not always clear why politicians push for “e-voting”. Turnout is really high in the Netherlands, compared to other countries. Without mandatory voting, in parliamentary elections on average 80% of the eligible voters will vote. As mentioned before, there might be a need for certain specific groups, such as visual impaired voters. However, no real study has been made on the size of this group or on other ways to improve the voting process for these voters. The most mentioned reason therefore remains the wish for speedy results after the polls close. Problematic is that in the debates in parliament, there is not always a clear distinction between “e-voting” and “e-counting”, so is it hard to tell if there would be a political will for “e-counting” of paper ballots.

The liberal parties do generally push for “e-voting” for voters living abroad, since they tend to receive the most votes from this group of voters. Steps have been made to improve the voting process for voters living abroad, but is it likely that the introduction of “e-voting (or in this case i-voting”) would increase the participation rate of voters living abroad.

5.3 Politics and “e-government”

Also on other areas of “e-government”, the current government seems to approach the topic with more hesitation than previous governments. The approach of the last two governments was to not only introduce new forms of “e-government”, especially with regard to communication between government and citizens through digital channels, but also to make it obligatory for citizens to use these channels (for example for filing taxes). In a vision paper presented in 2013 to the Parliament, the then Dutch government set the following goals. The provision of on-line services by the government must be improved. By 2017 at the latest citizens and businesses should be able to do any and all business they have to do with government bodies on-line. Examples are applying for a permit or lodging an objection. Off-line alternatives remain available. The government comprises all the central and local government bodies (municipalities, provinces and regional water authorities). The objective is to enable citizens to contact the government faster and more easily and to make it possible for them to do business at the time and place that suits them best. Another principle laid out in this vision paper is the strengthening of the position of citizens as a countervailing power to an increasingly interconnected “e-government”. By means of digital tools, citizens should be allowed to verify how they are registered, which organisations are using their data, and to correct their personal data if it is incorrect.

From the previous, it is clear that the previous government was striving for a system where most citizens would only interact with the government via digital ways. This has led to critique from different independent bodies in the Netherlands, most notably from the National Ombudsman. Also, one of the government’s independent scientific councils published a study which made it clear that there is a significant part of the Dutch population which is not capable to meet the demands that result from mandatory digital communication with governmental agencies [15]. Most notably from the study was that these citizens are not necessarily older, less technologically advanced citizens as was always assumed, but also younger, highly educated people. This has led to a reconsideration of the mandatory prescription of digital communication.

5.4 Political dimension conclusion

Overall, on the political dimension the Netherlands clearly scores lower on the Index than before 2006. However, the number of debates since then that were held on this topic do indicate that there is still a political will to re-introduce “e-voting”, even if this will is less unanimous than it used to be.

6 Law dimension

The Election Law of the Netherlands doesn’t contain any articles on “e-voting”. The articles that used to make it possible to use voting computers were removed after the events of 2006. The law is also written in a very technological dependent manner because the process of paper ballot voting is prescribed in great detail at the highest level of legislation. This means that it would not be easy to amend the legislation to include forms of “e-voting” without have to redraft the entire law or give a lot of discretionary room to the government in lower legislation. Given the discussions after 2006 and the problems that were discovered by the parliamentary committees [16, 17], it is unlikely that Parliament wouldn’t want to be closely involved in case of re-introduction of “e-voting”. However, since the Netherlands does have previous experience with “e-voting” and how to translate this into legislation, there is still existing knowledge that could be used if a re-introduction of “e-voting” would be considered. Overall, the score on the Law dimension would currently not be very high, but on this dimension it could be foreseen that it would be fairly easy to change this.

7 Technological dimension

As stated in paragraph 5, the current government is taking a slightly more cautious approach to “e-government” then the previous ones. However, in the Netherlands there already exists a pretty good technological support system for any form of “e-voting”. This technological infrastructure is good; all areas of the country have excellent power supply and coverage of internet and wifi is almost perfect. There is a national register of all citizens that is used to automatically produce a register of eligible voters. Everybody older than 14 has to have an ID card. Although the ID cards currently are not suitable for digital identification purposes, a bill is just been introduced in the legislative process to change this. If the bill is passed, all ID cards will be usable for online identification. Besides this form of identification, there is a system with a digital ID that citizens currently use to log in to governmental services. It also functions as a digital signature for different areas, such as tax returns. The government is working on improving the digital signature system by allowing private companies such as banks to expand their own online log-in systems for usage in communication between citizens and governmental organizations. The Netherlands has not played a very active role in the development of the new guidelines on "e-voting" of the Council of Europe, but because of the previous use of "e-voting", there is awareness of these types of standards.

Overall, the technological dimension should not be considered to be an obstacle in any way for a re-introduction of "e-voting" in the Netherlands.

8 Societal dimension

8.1 Internet penetration and mobile phone use

When it comes to internet penetration and mobile phone use, the Netherlands scores very high. The Netherlands is among the top EU 28 countries with the highest level of internet access at home. In 2017, 98 percent of Dutch households had internet access against a European average of 87 percent. In terms of high-speed broadband connectivity as well, the Netherlands ranks at the top. Nearly all Dutch households have a broadband connection at home.[2] In 2017, the Netherlands scored highest together with Sweden within Europe in terms of internet use on mobile devices, outside home or work, namely 87 percent. On average, 65 percent of the EU population aged 16 to 74 used the internet on their mobile devices. The use has grown rapidly in the Netherlands; in 2012, only 55 percent of the Dutch were mobile internet users. In 2017, 84 percent of people in the Netherlands used a smartphone outside their home or work. On average, in the EU, only 63% of people use a smartphone. Smartphone penetration was much lower in the Netherlands two years earlier in 2015, when 71% of the Dutch people used a smartphone. Laptops, notebooks and tablets were used by 54 percent of the Dutch population in 2017.

When looking at the amount of internet usage in the Netherlands, this is also very high. This graph shows the daily internet usage rate (for personal reasons) of online users in the Netherlands in 2017, sorted by age group. During the survey period, it was found that 99 percent of internet users between the ages of 25 and 34 were accessing the internet every day, but also that even within the group that used the internet the least, people of 55 and older, 88% uses the internet daily.

[2] This is based on an analysis of Eurostat figures by Statistics Netherlands (CBS).

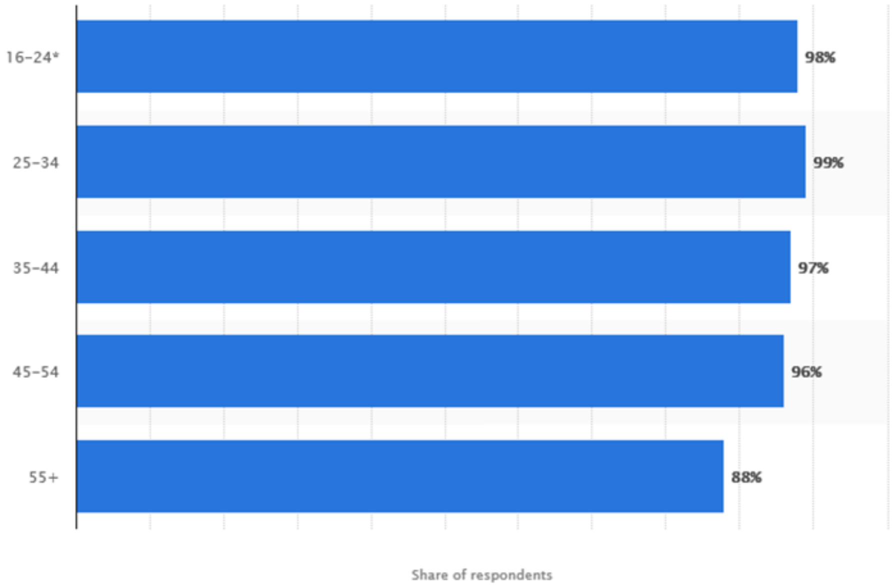


Fig. 1. Daily internet usage per age group

Studies also show that the use of internet by Dutch citizens is varied. In 2016 85% reported that they use online banking. 76% of citizens use internet for interaction with public authorities, for example for online self-service. Finally, 74% uses internet for online shopping.

8.2 Attitudes toward elections

Dutch citizens are active when it comes to participation in elections. Turnout for parliamentary elections is usually between 75 and 80%. Even though the turnout rate for the other types of elections (European Parliament, municipal and provincial) is significantly lower, there seems to be no real indication of the trend of decreasing turnout that can be witnessed in many other European countries.

Dutch voters also express a very high level of trust in the integrity of the election process. As figure 2 shows, from the data of the Dutch Parliamentary Election Study 2017 it becomes clear that most voters feel that the election process is either fair or pretty fair.

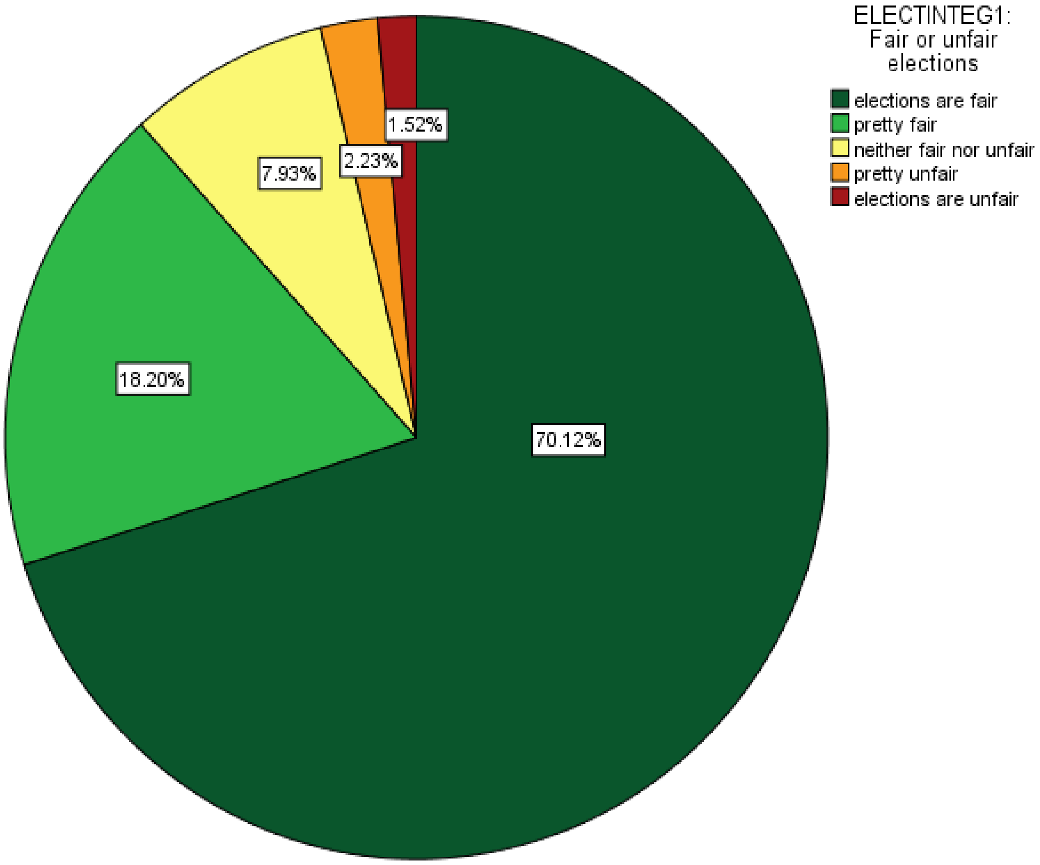


Fig. 2. Trust in the election process

8.3 Attitudes towards "e-voting"

To test the attitudes within Dutch society with regard to "e-voting", two questions were asked in the Dutch Parliamentary Election Study 2017. First people were asked which voting method they would prefer. Figure 2 shows that a small majority prefers to use paper ballots. This is somewhat remarkable because in similar studies done in 2006 and 2010 the majority still preferred the voting computer [18].

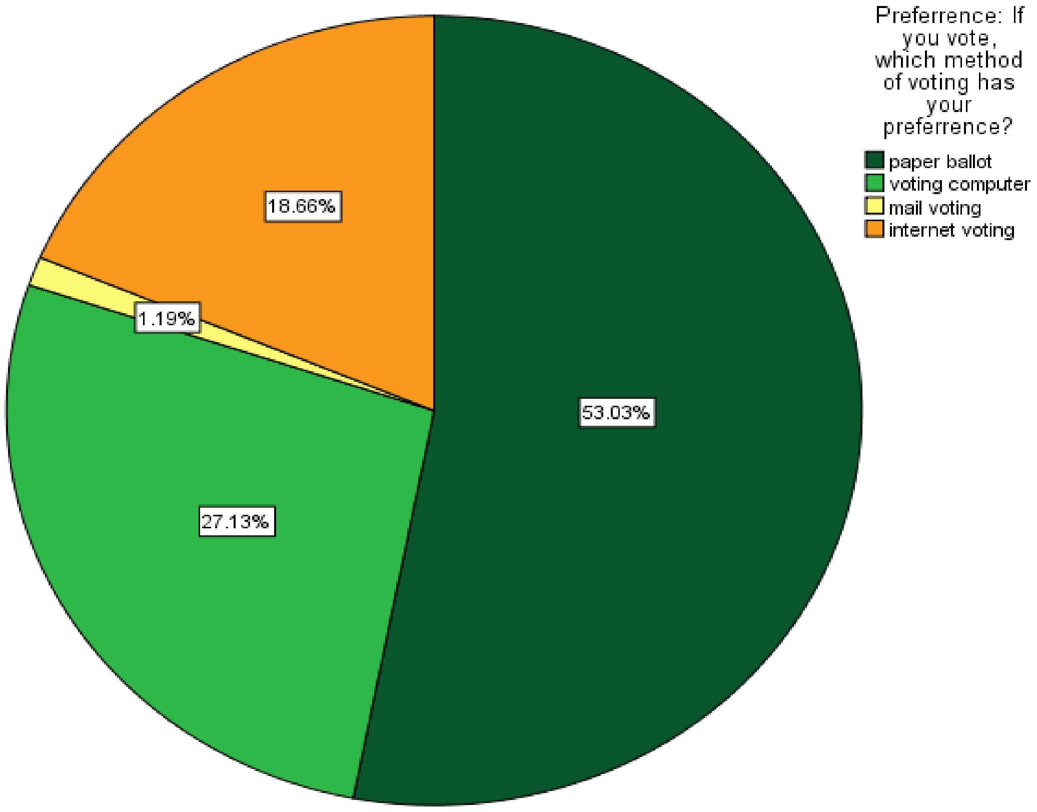


Fig. 3. Preferred voting method

Next, people were asked which voting method they would consider the most reliable. Figure 3 shows that almost 2/3 of the respondent feel that voting by paper ballot is the most reliable. Compared to the results mentioned above about the preferred method, this means that even though people do not feel that voting by voting computer is the most reliable, some of them would still prefer this. This difference in appreciation between preferred and most reliable method is even greater when it comes to internet voting; 18.1% of the respondents prefers this method, whereas only 6.2% feel this is the most trusted method. In these cases, convenience of the voting method seems to prevail over the question of trust.

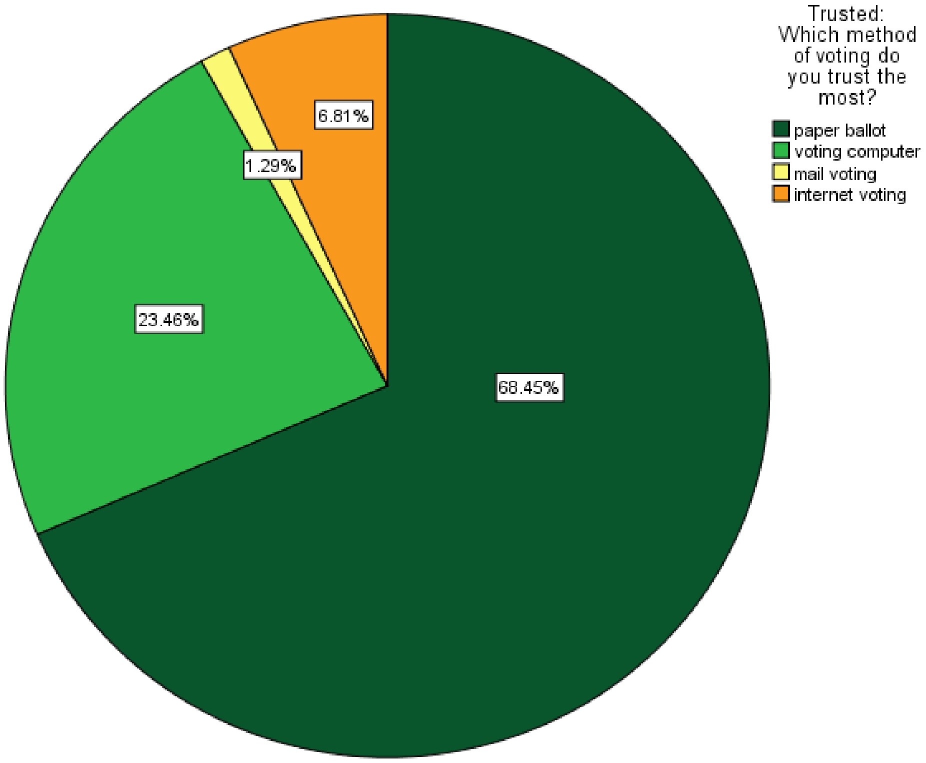


Fig. 4. Most trusted voting method

Often in debates concerning “e-voting”, it is claimed that this is a more modern way of voting and that younger, newer voters expect to be voting by means of technology. Therefore it is interesting to see if there is a difference in appreciation of voting methods between different age groups. In figure 4 the results are shown. For the Netherlands, the argument that younger voters would prefer more technologically advanced voting methods could not be used, based on these findings. Especially the younger voters have a relatively high preference for voting through ballot paper and mail. It is the older voters that express a preference for voting computers. Due to the history of “e-voting” described above, where the Dutch voters used voting computers on a large scale between 1970 and 2006, this result might be easily explained. The older voters have use voting computers for most of their elections and are therefore less experienced with paper ballot voting then younger voters. The voters between 25 and 34 express the highest preference for internet voting of all the age groups.

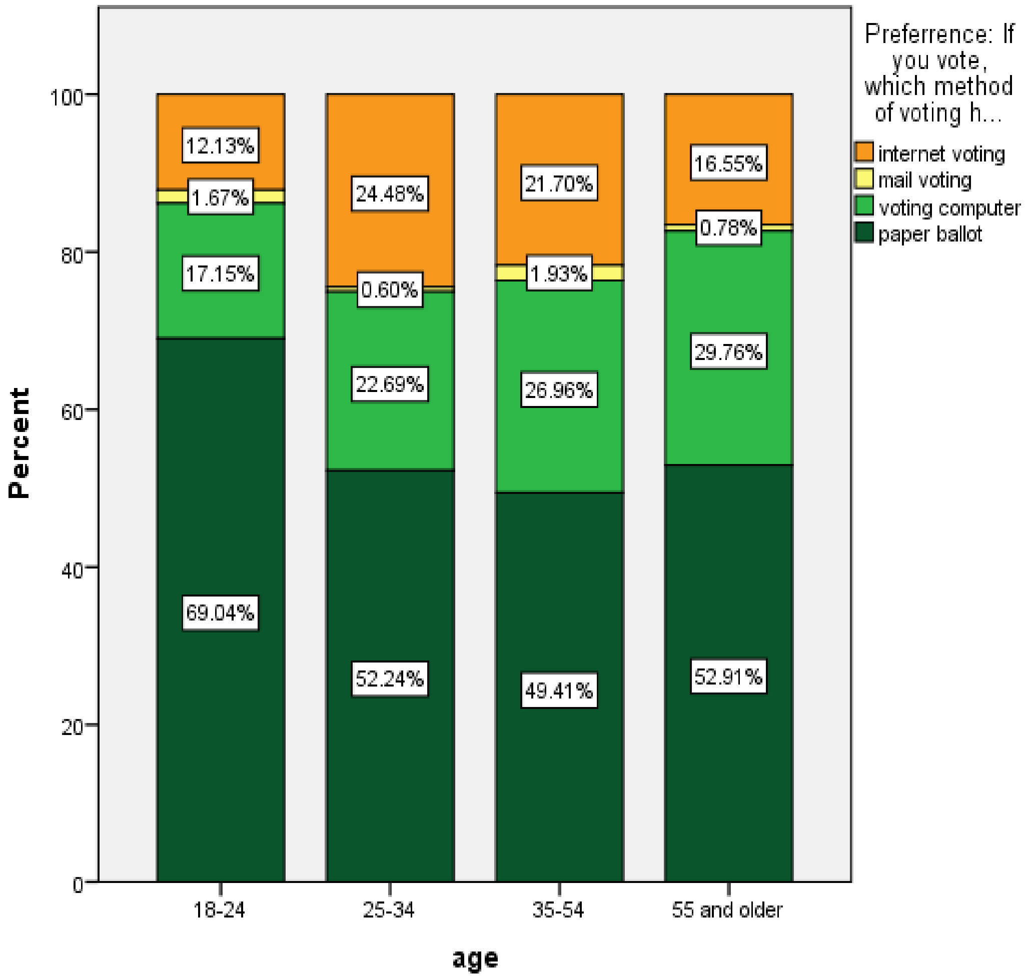


Fig. 5. Preferred voting method per age group

Finally, the role of age with regard to the trust in voting methods was examined. Figure 5 shows the results. Again, it should be noted that the youngest voters, those between 18 and 24 express the highest levels of trust in voting methods that are often described as being ‘old fashioned’ such as ballot papers and voting by mail. Compared to the other age groups these very young voters express the least trust in voting through voting computers or internet. The relative high level of trust in voting by mail expressed by this group is remarkable since they probably have the least day to day experience with the use of regular mail. As with the preferred method of voting, the voters between 25 and 34 express the highest level of trust in voting methods that use technology, such as voting computers and internet voting.

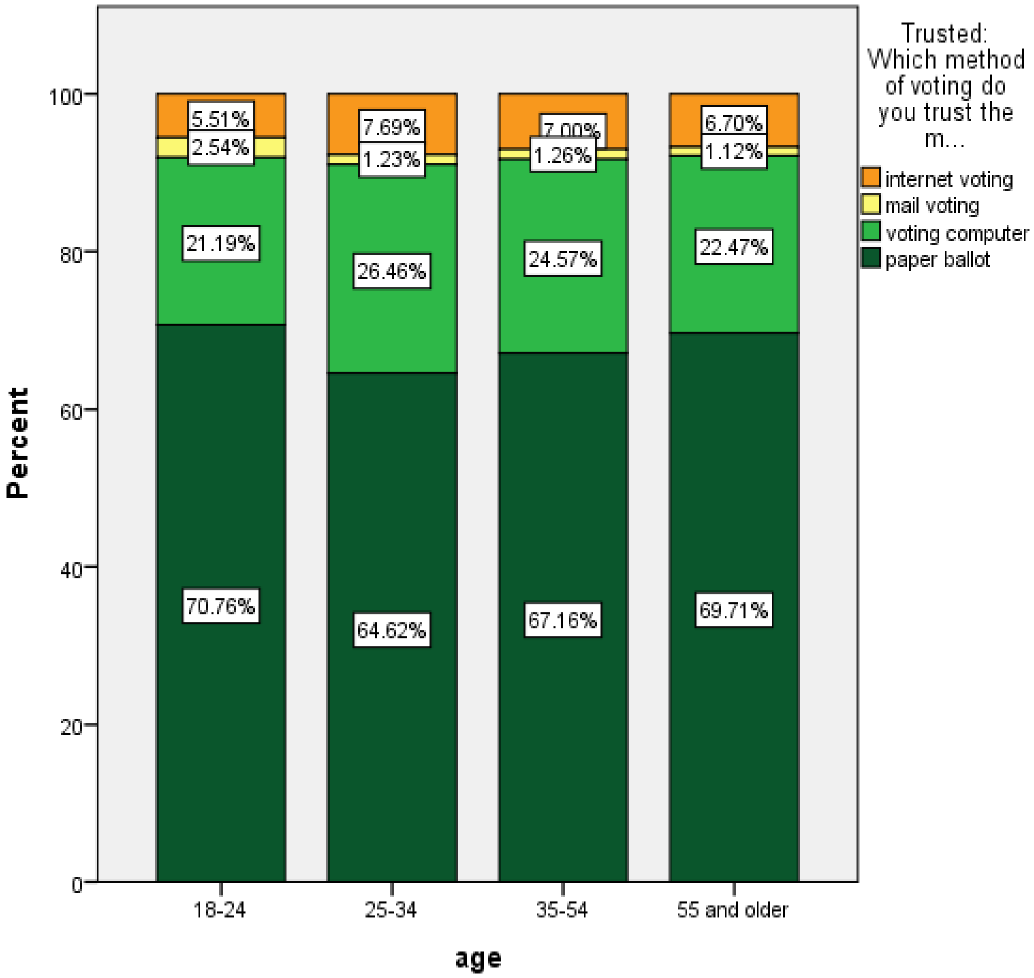


Fig. 6. Most trusted voting method per age group

8.4 Societal dimension conclusions

It is slightly difficult to determine how the Netherlands should be scored on the Index when it comes to the societal dimension. On the one hand, the country has a very high level of use of technological devices in daily life. Most people use smartphones and internet every day for a lot of different transactions. Also, participation in elections is high as well as the trust in the election process. However, the current attitudes of the voters towards new forms of use of technology in the voting process (voting computers and internet voting) show a preference for and a higher level of trust in paper ballot voting and even mail voting. Since it is especially the younger voters who express these attitudes, it seems unlikely that there will be a big push from society in the coming years to re-introduce “e-voting”.

9 Conclusions

The position of the Netherlands on the different dimensions of the Index score a mixed image. On the one hand, there is a lot of movement on the “e-government” domain that could be useful for “e-voting” services. Also, there is previous experience with “e-voting” and a wish coming from municipalities to return to “e-voting”. Even though the legal provisions currently do not account for the possibility of “e-voting”, they have done so in the past and those previous provisions could be re-introduced. The technological dimension shows no problems for “e-voting” in the Netherlands. However, the societal dimension is more difficult. The attitudes of voters towards new voting technologies do not match with their willingness to embrace modern technologies in other aspects of their lives. As long as the trust of Dutch voters in voting computers and internet voting remains at the current low level, it might not be feasible to reconsider the abandonment of “e-voting”. Finally, there seems to be a lack of a clear problem within Dutch elections that can only be solved by (re-)introduction of “e-voting”. In light of the problems that arose with “e-voting” in 2006, there will be most likely need to be a pressing need for “e-voting” before the political and societal will to use it again could lead to a change in the current situation.

References

1. Loeber, L. The use of technology in the election process: Who governs? Forthcoming
2. Loeber, L. E-Voting in the Netherlands; from General Acceptance to General Doubt in Two Years. 3rd International Conference on Electronic Voting 2008.
3. Oostveen, Anne-Marie. "Outsourcing democracy: losing control of e-Voting in the Netherlands." Policy & Internet 2.4 (2010): 201-220.
4. Jacobs, Bart, and Wolter Pieters. "Electronic Voting in the Netherlands: from early Adoption to early Abolishment." Foundations of security analysis and design V. Springer, Berlin, Heidelberg, 2009. 121-144.
5. Barrat, Jordi, and Carlos Vegas. "Overview of Current State of E-Voting Worldwide." Real-World Electronic Voting. Auerbach Publications, 2016. 67-92.
6. Prosser, Alexander, and Robert Krimmer. "The dimensions of electronic voting technology, law, politics and society." Electronic Voting in Europe Technology, Law, Politics and Society (2004): 21-28.
7. Krimmer, R., & Schuster, R. The E-Voting Readiness Index. 3rd International Conference on Electronic Voting 2008.

8. Lufi Herawan, Dana Indra Sensuse, Puji Rahayu, Csf for Implementation E-voting model: A systematic review.
9. Emaase, Patrick Mokodir, and Evans Miriti. "E-voting readiness in Kenya: A case study of Nairobi county."
10. Hapsara, Manik. "E-voting Indonesia: framing the research." Information Systems and Technologies (CISTI), 2014 9th Iberian Conference on. IEEE, 2014.
11. Kumar, V., Mukerji, B., Irfan, B. and Ajax, P. (2007) Factors for Successful e-Government Adoption: A Conceptual Framework, The Electronic Journal of e-Government, 5, 1, 63-77.
12. United Nations Report, (2008) UN E-Government Survey 2008: From E-Government to Connected Governance, ISBN 978 -92-1-123174-8, UN White paper, http://unpan1.un.org/intradoc/groups/public/documents/UN/UNPAN028607.pdf.
13. Schmeets, H. and R. Van der Bie (eds.) (2008) Het nationaal kiezersonderzoek 2006, Opzet, uitvoering en resultaten (Voorburg: Centraal Bureau voor de Statistiek).
14. Loeber, Leontine. "E-voting in the Netherlands; past, current, future." Proceedings of the 6th international conference on electronic voting (EVOTE). TUT Press, Tallinn. 2014.
15. Wetenschappelijke Raad voor het Regeringsbeleid, "Weten is nog geen doen. Een realistisch perspectief op zelfredzaamheid." (2017).
16. Hermans, L. et al.: Voting machines, an orphaned subject, Report by the Advisory Committee regarding the decision making process for voting machines, April 17, 2007.
17. Korthals Altes, F. et al.: Voting with confidence, Report by the Election Process Advisory Comission, September 27, 2007.
18. Loeber, Leontine. (2011). Voter trust in the Netherlands between 2006 and 2010. CeDEM11 Proceedings of the International Conference for E-Democracy and Open Government. 323-334.

Online Voting in Indigenous Communities: Lessons From Canada

Nicole Goodman[1](✉) [0000-0002-8607-2595], Chelsea Gabel[2] [0000-0002-1007-5351], and Brian Budd[3][0000-0003-3554-430X]

1 Department of Political Science, Brock University, St. Catharines, Canada
nicole.goodman@brocku.ca

2 Department of Health, Aging and Society and Indigenous Studies Program, McMaster University, Hamilton, Canada gabelc@mcmaster.ca

3 Department of Political Science, University of Guelph, Guelph, Canada
buddb@uguelph.ca

Abstract. Most studies of online voting examine adoption at national and subnational levels or among municipal governments. Very few examinations, however, focus on implementation in Indigenous communities. Drawing on community-engaged survey work with three First Nations in Canada – Tsuut'ina Nation, Wasauksing First Nation and Whitefish River First Nation, 28 interviews with Indigenous leaders, identified experts, online voting vendors and federal government representatives as well as a focus group, we examine why Indigenous communities in Canada are drawn to online voting, who is using it, potential impacts on participation, and good practices that can be learnt from these experiences. Our findings suggest broad support for online voting and satisfaction from Indigenous voters. Though online voters tend to be older, educated, wealthier and live off reserve, survey results indicate online ballots could engage some Indigenous electors to vote more frequently. Notably, we find that online voting is a critical tool to reach and engage off reserve citizens. Finally, we outline a number of good practices for online voting deployment that fall into four themes: (1) community knowledge and engagement (2) tools and strategies, (3) clear processes and resources, and (4) a focus on technology.

Keywords: Online voting, First Nations, Canada, Community-engaged research

1 Introduction

Online voting has been used in a number of contexts in jurisdictions around the world. Most studies examine adoption nationally [26, 30, 32], sub-nationally [29] or in the context of local governments [10, 15, 24]. Very few examinations, however, focus on online voting implementation in Indigenous communities [8, 9]. Consequently, we have a modest understanding of the effects online voting can have on Indigenous communities and the lessons that can be learnt from their experiences. Filling this lacuna is important for scholarly understandings of whether online voting can be leveraged to overcome some of the barriers to voting Indigenous peoples face, including reduced voter access and registration challenges [4]. Online voting also has enormous practical importance as Indigenous peoples around the world pursue adoption at growing rates in countries such as Canada, the United States and New Zealand [3, 8]. Lessons from Indigenous experiences with online voting are especially valuable for remote and rural communities, where voters often face similar accessibility challenges because of limited broadband infrastructure and weaker digital literacy.

This article uses a community-based research approach with three First Nations in Canada: Tsuut'ina Nation, Wasauksing First Nation and Whitefish River First Nation. The methods include surveys with Indigenous voters, 28 semi-structured interviews with Indigenous leaders, identified experts, online voting vendors and federal government representatives, and a focus group, to understand the effects of online voting in Indigenous votes. Drawing on this rich collection of research we examine why Indigenous communities use online voting, who votes online and why, impacts on participation, and good practices that can be taken away from these experiences. Our findings indicate there is broad support for online voting and satisfaction from Indigenous voters. Though online voters tend to be older, educated, wealthier and live off reserve, survey results suggest online voting could engage some Indigenous electors to vote more frequently. Notably, we find that online voting is a critical tool to reach and engage off reserve citizens. The importance of reaching these electors is crucial given Canada's multi-level governance structure and the fact that in some cases the federal government imposes quorums that Indigenous communities must meet to pass votes. The article also presents a number of good practices to guide the implementation of online voting in Indigenous contexts.

The article proceeds in six sections. First, we summarize the relevant scholarly literature. This includes a brief historical summary of colonial governance in Canada and relevance for online voting adoption as well as a summary of literature on Indigenous adoption of digital technologies and online voting. Next, we provide a summary of First Nations we have worked with and whose experiences are reflected in this article. Fourth, we review our approach and data. Fifth, we present an overview of the survey data collected through the project and good practices that were developed from four years of community-engaged work, 28 interviews and a focus group. Finally, we conclude with a summary of results and suggest opportunities for future research.

2 Literature

Though studies explore the effects of online voting in the context of national [26, 30, 32], sub-national [29] and local governments [10, 15, 24], there is a lacuna in research examining deployment of online ballots in Indigenous communities. While projects studying Indigenous adoption of online voting are underway in New Zealand, where more than 90 iwi organisations use the technology, these findings have yet to be published. Early work with First Nations in Canada is presently the only published material that examines the impacts of online voting on participation and governance in Indigenous communities [8, 9]. This section provides historical context regarding the structures of governance in Canada and summarizes the findings of existing research.

2.1 Historical Governance Structures

The adoption of online voting by Indigenous communities in Canada is linked to a history of settler colonialism and ongoing resistance enacted by Indigenous peoples to reassert local autonomy and political self-determination. Following the passage of the Canadian constitution in 1867, steps were taken to consolidate territory by dispossessing Indigenous peoples of their traditional lands and reconstituting Indigenous governance structures under the terms set out by federal legislation [7]. These approaches to limit the territorial presence and governance capacity of Indigenous communities soon evolved into outright attempts to eliminate Indigenous peoples as distinct political and social collectives through violent and assimilative policies designed to forcefully integrate Indigenous peoples into mainstream Canadian society [31]. While Indigenous resistance to these attempts has been constant throughout Canadian history, the beginning of the 1970s marked a shift in Indigenous-State relations as Indigenous communities and politicians successfully expanded the recognition of Indigenous rights to self-determination through intergovernmental negotiations, the inclusion of Aboriginal rights in Canada's constitution and ongoing acts of organized protest at varying levels of governance [19, 23]. While Indigenous communities still face many social, political and legal challenges toward the realization of self-government and by extension self-determination, Indigenous governments have assumed a growing number of jurisdictional responsibilities while exercising greater control over the design and delivery of local policies and services. It is within this tension between settler colonialism and Indigenous resistance that we can understand and study the adoption of digital technologies by First Nations and other Indigenous bodies.

2.2 Indigenous Adoption of Digital Technologies

Digital technology and infrastructure present unique opportunities for Indigenous communities to address social and political challenges while also strengthening local governance capacity. There is a small, yet growing literature examining the adoption of digital technologies by Indigenous communities. Studies focus on the introduction of technologies in the areas of healthcare, education, social services, economic development, and cultural renewal addressing the ways in which technology can strengthen administrative capacity and overcome challenges in local service delivery [18, 25]. Others point to challenges faced by Indigenous communities in Canada interested in adopting digital technology such as digital divides. McMahon et al. [20], for example, highlight this issue by exploring the barriers remote and rural Indigenous communities face with respect to broadband infrastructure and internet services given the marketization of internet service delivery. While First Nations have proven resourceful in responding to these challenges through the development of community-based approaches to improve internet access [see 20], they continue to face structural barriers to adoption due to geographic remoteness and a lack reliable broadband infrastructure.

Despite these issues, digital technologies have become an important tool not only for the improvement of local services, but also for the pursuit of broader political goals related to self-determination. Scholars have begun to consider the implications for the adoption of digital technology for self-determination, developing concepts like "digital self-determination" [21] and "digital decolonization" [1] as a way to interpret the varied empirical examples of technological adoptions by Indigenous communities. These concepts focus on the ways in which technology can contribute to the realization of Indigenous self-determination by facilitating the decentralization of political authority and the development of governance capacity.

2.3 Indigenous Deployment of Online Voting

To date, the only published research of Indigenous use of online voting examines experiences in Canada [8, 9], albeit projects working with iwi, Te Rūnanga o Ngāti Awa and others are underway in New Zealand [3]. Research with First Nations in Canada has explored the rationales for online voting adoption, benefits to participation and governance, as well as legal, social and political challenges. All of these contributions, however, focus on examining individual First Nations and do not provide comparative assessments.

Research shows online voting fosters and strengthens community connectedness among Indigenous peoples [5, 9]. Colonial legacies have left many Indigenous persons disconnected from the political processes in their communities, especially those living off reserve lands. Online voting has been found to alleviate political alienation by generating dialogue and engagement between government and citizens, notably among youth and elders [9]. In the context of Wasauksing First Nation, Budd et al. [5] find that online voting connects off reserve members to discussions of community business and policy, and is a critical tool for enhancing participation to meet the quorums necessary for critical votes. More generally, the excitement and novelty of online voting has helped First Nations complement traditional in-person engagement strategies and strengthened administrative capacity [5]. In this way, online voting has been shown to make both direct and indirect contributions to community connectedness in First Nations.

Further, online voting makes meaningful contributions to community modernization and improve the self-governance capacity of First Nations. Interview research with First Nation officials and administrators found that First Nations are increasingly looking for ways to update processes and procedures to reflect the current realities of members' day-to-day lives. Online voting is often adopted as part of a broader suite of digital tools designed to modernize First Nation governance [5, 8]. It is also closely linked to gains in self-governance capacity in the form of improved administrative capacity and enhanced ability to develop and pass community legislation. Specifically, research finds online voting improves First Nations' administrative capacity by simplifying ballot tabulation processes and providing immediate results [8]. This improved capacity has positively influenced trust between citizens and government and increased confidence in voting results [5]. More importantly, online voting strengthens Indigenous self-government by enhancing the capacity of Indigenous communities to enact their own laws and regulations [5]. As mentioned, the ratification of community legislation requires First Nations to reach a minimum quorum of community participation. By fostering the participation of off-reserve members, online voting serves as an important tool to reach a level of participation that would otherwise would be difficult to secure through traditional voting methods alone. The ability to develop and pass legislation is an important symbolic and practical consideration in the pursuit of self-government, helping to peel back layers of colonial legislation while allowing First Nations to assume greater jurisdiction over their affairs.

Despite positive experiences, enhanced community connectedness and self-governance capacity, research also points to key challenges with Indigenous implementation of online voting. Many of these challenges stem from the lack of reliable, high-speed internet access within First Nations. Unfamiliarity with, and difficulty navigating, online voting registration procedures is also a barrier. Gabel et al., [8] point out that a two-step voting process, which requires citizens to register to vote online before accessing an internet ballot, is viewed by some as being too complicated and burdensome and discourages uptake among community members with less familiarity and experience using computers and the internet. Conflicts have also been noted between online voting and traditional Indigenous values and customs. Community members have also communicated concern that online voting and growing digitization of governance will lead to less face-to-face dialogue [5, 9]. The ability to openly discuss issues and decisions in-person is a key component of decision-making procedures within many Indigenous cultures. While these concerns have been expressed by some First Nation members, research has observed that when human connections are prioritized and online voting is integrated as complementary, rather than a replacement for traditional forms of community engagements and decision-making, satisfaction with online voting is high [5].

Finally, as noted above, the most significant impediment to online voting use is legislative in nature. While a growing number of First Nations have opted out of the provisions of the Indian Act and First Nations Elections Act by developing and ratifying their own electoral codes or self-government agreements, many continue to face legislative barriers to choosing the voting methods they will use in their elections [22].

In sum, existing research into online voting use in First Nations demonstrate its alignment with the broader political goals of building community capacity and enacting self-determination. While questions and concerns linger about the cultural appropriateness of internet ballots and working with private-sector vendors, there is an emerging relationship between online voting and self-determination. Absent in the literature on online voting use in Indigenous communities are generalizable insights into good practices that can facilitate the successfully deployment of online voting in Indigenous contexts as well as comparative analyses comparing findings across First Nations. This article addresses these gaps by reflecting on the experiences of three First Nations and providing a set of good practices for the deployment of online voting in Indigenous communities.

3 History and Context: Online Voting Developments in Indigenous Communities in Canada

Indigenous adoption of online voting is growing in Canada. Of the three groups of Indigenous peoples in Canada: First Nations, Métis and Inuit, the bulk of online voting activity has occurred in First Nations. Talthan First Nation in British Columbia was the first to use online voting in 2011 for the ratification of two agreement votes. Positive reviews of deployment spread and soon other First Nations in British Columbia carried out pilots. Huu-ay-aht First Nation adopted online voting in 2012 for its general assembly and Squamish First Nation used it to support a referendum on its membership agreement in 2013. Since then, online voting has been used in more than 80 of the 634 First Nations in Canada across six provinces: Alberta, British Columbia, Manitoba, Newfoundland, Nova Scotia, Ontario and Quebec, typically for ratification or agreement votes [22]. Use of online voting in Indigenous votes would be more frequent were it not for Canada's multi-level governance structure, whereby the Indian Act and First Nations Elections Act, legislation written by the federal government, governs elections and referendums unless a First Nation has opted out. Regulations of these acts outline the ballot methods that can be used and currently only provide for paper voting.

While a majority of First Nations have taken control of their elections by passing self-government agreements or custom codes for the specific governance of elections, 225 First Nations fall under federal legislation, see Table 1, and therefore cannot use online voting for elections and referendums. These nations can, however, deploy online voting for other types of votes such as ratification votes or the passage of framework agreements such as Land Codes [22].

Table 1. Governance structure of First Nations' elections in Canada[1]

Legislative framework	Number of nations
Indian Act election system	174
First Nations Elections Act	51
Custom election codes	353
Self-government agreements	40

Interestingly, among First Nations that fall under federal legislation, online voting is often adopted in contexts where a federally mandated quorum must be met to pass community-developed legislative frameworks. For example, there is a 25 percent quorum to pass a Land Code agreement, which allows the nation to regain control of their lands from the federal government. In some cases about 75 percent of a nation's citizens can live off reserve, making meeting quorum challenging with traditional paper voting alone. As such, improving voter participation and access among those

[1] This table is updated and adapted from an earlier paper by Goodman and Pammett [13] and reflects election governance structures as of May 11, 2018 according to federal government records The number of nations accounted for is 618, meaning there is not up to date information for 16 nations.

living off reserve is a key motivation for online voting adoption. Greater youth engagement, expedited ballot tabulation and other administrative efficiencies are other primary rationales [8].

Finally, online voting is typically adopted as a complementary method of voting. In some instances where internet connectivity is poor and access is limited, telephone voting is also offered to enhance voter access. A few remote and rural communities have eliminated paper voting because it was hardly used and opted for fully electronic elections.

4 Community Profiles

This article incorporates research findings from three First Nations in Canada. Two of the communities, Whitefish River First Nation and Wasauksing First Nation, are located in northern Ontario while the third community, Tsuut'ina Nation, is located in Alberta. All three have deployed online voting to ratify community-based legislation. While the legal context of these votes is beyond this article, in general, when First Nations wish to opt-out of federal legislation, they must develop and ratify legislation to replace it. As noted above, this ratification process typically involves meeting a federally mandated quorum that includes a minimum threshold of participation.

Whitefish River First Nation is an Ojibway community with a total registered population of 1336 members, 900 of whom reside off reserve. The community used online voting in 2014 to ratify Matrimonial Real Property legislation. Wasauksing First Nation is an Ojibway, Odawa and Pottawatomi community with 1341 members, 912 of which live off reserve. Wasauksing's deployment of online voting took place in 2016 and was used to ratify community-developed Land Code legislation. The third community, Tsuut'ina Nation, is the largest of the three with 2132 members with roughly half of those members residing off reserve. Tsuut'ina deployed online voting to ratify its Election Code, in a Chief and Council election, as part of a referendum on the production and sale of cannabis within the nation and for an opinion poll.

5 Approach and Data

Indigenous peoples have a history of being the most researched people in the world. This research is often undertaken without sufficient permissions being obtained or community consultations [28]. Socially-engaged research approaches, which focus on carrying out research with Indigenous peoples rather than on them are growing in popularity since they are more inclusive and often involve active community partnerships that produce knowledge outcomes which directly benefit them [22]. Research undertaken for this article takes a Community-Engaged Research (CER) approach, which falls within the spectrum of socially-engaged research approaches. CER focuses on promoting research partnerships that are based on empowerment, respect and inclusiveness, and work to balance existing power inequities between researchers and communities [16]. Adopting this framework means that partners are treated as equal members in all phases of the research process and share control over the research

[17]. This approach not only ensures the production of valuable knowledge outcomes for community actors, but also enhances the depth and breadth of research questions and better informs scholarly outputs with Indigenous knowledge. It also has the added benefit of contributing to, and influencing, social change.

Since May 2014 our research project, First Nations Digital Democracy, has employed a CER approach to work collaboratively with several First Nations in Canada located in the provinces of Ontario and Alberta. Closer examinations of the findings from these communities - Tsuut'ina Nation, Wasauksing First Nation and Whitefish River First Nation – are presented in this article. Each of these First Nations used online voting during the project and partnered with us to better understand the impacts of online voting on participation and governance. The type and nature of data collected varied by community based on their unique needs and input into the research process.

Data from Wasauksing First Nation was collected from online and paper voters during a Land Code ratification vote. Online voting was available from 9:00am EST December 10, 2016 until 8:00am EST on February 25, 2017, which was the official voting day. Only paper ballots were available after 8:00am on the 25th. Once voters had cast an online ballot they were prompted to take a survey about their voting experience. Paper voters were approached at the polls by project personnel and had the option to complete exit surveys on iPads or paper. Youth and elders in the community were trained and hired to support recruitment. A total of 29 online voters took part in the survey, with 15 respondents (N=15) completing the full survey for a response rate of 20 percent. Sixty-six paper voters completed a survey (N=66), for a response rate of 66 percent.

The Whitefish River First Nation vote took place from March 2 to 6, 2015 as part of a ratification vote on Matrimonial Real Property. Voting from March 2 to 5 was carried out entirely online, while March 6 was paper voting only. In this community surveys were carried out with paper voters only. A total of 123 surveys were completed (N=123), which represents a response rate of 81 percent. The same community-engaged approach was taken wherein youth and elders were hired to support survey recruitment on the official voting day.

Finally, Tsuut'ina Nation carried out an opinion poll during a two-day community meeting on April 17 and 18, 2018 regarding whether to allow the production and sale of cannabis on the nation and to determine which voting methods (paper, online or paper voting with electronic tabulators) should be used for elections and referendums. In an effort to build digital literacy in the community, online voting was the only available method to participate in the opinion poll. Research project members were on hand with iPads to walk citizens through the voting process and survey. A total of 139 voters (N=139) completed the survey (out of 155 that cast a ballot in the opinion poll) for a response rate of 90 percent.

These samples are self-selected and quite small. While in some cases response rates are 80 percent or higher, contextual circumstances and attitudes in First Nations vary greatly across communities. These considerations prevent us from conducting deeper analysis and cause us to use caution in drawing conclusions about the representativeness of the results for all First Nations in Canada. That said, this data is the first of its kind and provides an understanding of attitudes toward, and satisfaction with, online voting, likelihood of use, concerns and past voting behaviour. The community-engaged research approach meant that the study was tailored to the needs of each individual First Nation, making the research design slightly different for each partnership.

The article also draws upon 28 semi-structured interviews carried out with Indigenous leaders and community actors, identified experts, online voting vendors and government agencies responsible for Indigenous affairs and elections in Canada. Interviews were conducted between December 2017 and April 2018 and were carried out as part of a project sponsored by Indigenous and Northern Affairs Canada to better understand good practices of online voting and policy recommendations for future deployment. Interview questions addressed Indigenous attitudes toward online voting, rationales for use, benefits and drawbacks of adoption, effects of online voting implementation and good practices and recommendations for future use. This research also included a focus group with four administrators from Tsuut'ina Nation conducted in March 2018. The focus group used the same guide as the interviews, but was conducted in a more interactive, focus group setting.

6 Findings

6.1 Satisfaction With and Willingness to Vote Online

Research carried out as part of the First Nations Digital Democracy Project finds broad support for the use of online voting in Indigenous elections and votes. In Wasauksing First Nations's 2017 Land Code ratification vote, for example, 30 percent of all ballots were cast online. The remaining 70 percent were cast by paper (40 percent) and mail-in ballot (30 percent). Similarly, in a 2016 Land Code vote in Metlakatla First Nation located in British Columbia, 48 percent of votes were cast online, 30 percent by mail and 21 percent by paper ballot at the polls. Each of these examples are cases where online voting was deployed for the first time and demonstrate that Indigenous voters are willing to make use of the voting method. In particular, the sizeable portion of votes cast remotely (by internet or mail ballot): 60 percent in Wasauksing and 79 percent in Metlakatla, highlight the importance of using remote voting methods in First Nations given the importance of engaging citizens living off reserve. Both aforementioned First Nations have sizeable off reserve populations and needed to meet the 25 percent quorum imposed by the federal government to pass their Land Codes. In the case of Wasauksing, for example, 66 percent of citizens live off reserve, whereas in Metlakatla, off reserve citizens account for 89 percent of membership. Interviews with Indigenous leaders across Canada echo the importance of leveraging technology to engage off reserve citizens. Similar observations about using online voting to improve voting access for electors that do not live on territorial lands have been made in the context of Canadian municipalities with large seasonal populations [14] and for expatriate voters in other countries [11].

Survey data from Wasauksing First Nation and Tsuut'ina Nation show high satisfaction with online voting and indicate voters would like to see it offered in the future for reasons of convenience and accessibility. Ninety-one percent of respondents in Tsuut'ina indicated that they were either 'very' or 'fairly' satisfied with the online voting process. Top reasons for using online voting included: convenience (36 percent), privacy (17 percent), wanting to try something new (17 percent), and accessibility (15 percent). Similar findings of support are present in Wasauksing First Nation. One hundred percent of online voters who completed our survey indicated they were either 'very' or 'fairly' satisfied with the voting method. Primary rationales for voting online included convenience (41 percent), accessibility (24 percent), and wanting to try something new (12 percent). Such rationales for use are consistent with the reasons given by voters in Canadian municipal elections where convenience has been shown to be the main motivating factor [14].

While the above examples illustrate that voters are willing to make use of the voting method and are satisfied with it, those who prefer to vote by paper are also supportive of the policy change. Where data is available, we find that having online voting as a complementary voting method is desired by paper voters. In Wasauksing First Nation, paper voter respondents were asked whether they would consider voting online in the future. Sixty-three percent said they would consider voting by internet for a future vote. Twenty-eight percent of these respondents said they would vote online 'in all circumstances', while 35 percent reported wanting to use it under 'special circumstances' such as in cases where being too busy, inclement weather, illness or mobility issues prevented them from attending a physical poll location. There were, however, 35 percent of respondents that said they would not vote online in future.

A similar survey carried out with paper voters in Whitefish River First Nation reveals comparable results. Fifty-six percent of paper voters indicated that they would vote online in the future (20 percent 'in all circumstances', 36 percent in 'special circumstances'), while 33 percent said they would not vote online. Though there are understandably electors who prefer to vote by paper and want to continue using that voting method, the fact that a majority of paper voters in these First Nations say that they would vote online in the future, particularly under 'special circumstances' where their participation may be limited without a remote voting option, suggests broad support for online voting.

Reasons why some paper voters may be hesitant to vote online likely have to do with concerns about internet access and online voting security. In Wasauksing First Nation a majority of paper voters said they did not have concerns about online voting (41 percent), however, 21 percent reported concerns about lack of internet access and 18 percent cited security as a concern. Other, less prevalent, concerns included: fraud (8 percent), privacy (8 percent), lack of computer and internet knowledge (2 percent) and loss of voting traditions (2 percent). Similarly, paper voters in Whitefish River First Nation expressed lack of a computer or access to the internet as a concern (26 percent), security (19 percent), the replacement of voting traditions (8 percent), privacy (6 percent), fraud (5 percent) and 'other' reasons (7 percent).

In both cases, concerns about lack of access to technology and the internet speaks to issues with broadband infrastructure and affordability. In some remote communities the cost of a monthly internet plan can often be 3 to 4 times the cost of a comparable plan in a suburban area. While every First Nation is different, these results suggest strong support for online voting among Indigenous voters who would use the service and those who would typically opt for a paper ballot.

6.2 Who Votes Online?

Data collected from Wasauksing First Nation allows us to compare the socio-demographic characteristics of voters who chose internet and paper ballots, and confirms that online voting seems to be especially appealing to those living off reserve. There is also evidence to suggest that online voting could motivate a modest portion of less frequent voters to take part.

Looking at the age of internet and paper voters who completed surveys in Wasauksing First Nation shows that online voters are middle-aged, with the largest group falling in the 45 to 54 age category. Online voters had a mean age of 48 years and paper voters 44 years. Although the online voter sample was small, only one person under the age of 35 who completed our survey chose to vote online. Younger voters were much more likely to cast a paper ballot. These findings are consistent with studies of online voting use by young people in municipal elections in Canada [14] and Norway [26], which suggest young voters are more likely to vote by paper given that it is often one of their first voting experiences and is seen as a symbolic and ceremonial act.

Interestingly, the oldest voters, those over the age of 55, were also more likely to vote by paper than internet. This likely has to do with older voters communicating greater concerns about lack of internet access and experience with computers. Thirty-six percent of paper voters remarked that they do not have internet access at home and a majority of this group (60 percent) were over the age of 55. Indigenous voters over the age of 55 were also much less likely to access the internet regularly. While online voting use in municipal elections drops off with age, it typically occurs over the age of 65. The fact that persons aged 55 years and older are much less likely to vote by internet likely has to do with the limited availability of internet and weaker digital literacy in First Nations.

Moving on to other socio-demographic characteristics, we see that online voters in Wasauksing First Nation report being more educated and having a higher household income than paper voters. Online voters are also more likely to live off reserve. Findings about online voters being more educated and having higher incomes than paper voters have also been observed in Canadian municipal examinations and suggests that the extension of online voting may be more about improving convenience for persons who were likely to vote anyway rather than attracting electors from all socio-demographic groups [12, 14]. In this context, however, higher education levels and reported income could be related to the greater education and employment opportunities located off reserve. Likewise, lower reported educational attainment and income for paper voters could be linked to living in the community. The finding about online voters being more likely to live off reserve makes sense, especially given the enhanced accessibility for these electors.

Table 2. Mean socio-demographic characteristics of voters in Wasauksing First Nation

Socio-demographic characteristic	Online Voters	Paper Voters
Age	48 years	44 years
Education	Completed technical, community college	Completed technical, community college
Annual household income	$60 000 to $79 000	$40 000 to $49 000
Marital status	Married	Married
On or off reserve	Off reserve	On reserve

Interviews with election administrators in Wasauksing First Nation and Indigenous leaders in other nations confirm the value of online voting for off reserve citizens. As one leader remarked, “Because our community is dispersed with 85% away from our homeland, and 60% a significant distance away from the homeland, online voting helps to further include everyone, and that’s quite important when it comes to thinking about it from First Nations’ perspective, off reserve specifically. First Nations that are not connected to their homelands or their cultures or their peoples, have kind of created for themselves a third party, so to speak. This would allow them to include themselves in important processes, cultural processes, electronically in some way, shape or form.”

Interviewees commented on citizens living in different cities, provinces and countries and the importance of engaging these members to ensure community voice is represented in the decision-making of the nation. While voting by mail is another remote voting option, many we spoke with communicated issues with delayed ballots, problems with the mail system, and electors simply not leaving enough time to mail a completed ballot as barriers that online voting can address. Online voting is clearly desired in Wasauksing First Nation and in other First Nations where a sizeable proportion of citizens typically live off reserve (places like the Mowhawk Council of the Awkwesasne which spans Ontario, Quebec and the United States and Metlakatla First Nation are other examples). In fact, in cases where a majority of citizens live off reserve, the adoption of online voting may be essential to engage residents especially in instances where quorum must be met such as in the case of Land Code and MRP ratification votes.

6.3 Potential for Engagement

Finally, does online voting have the potential to engage Indigenous voters? The Tsuut’ina Nation survey asked respondents whether they would have taken part had online voting not been an option. A majority of participants indicated that they ‘definitely’ or ‘probably’ would have participated in the opinion poll regardless of whether online voting was used. However, 9 percent indicated that they ‘definitely’ or ‘probably’ would not have voted had the voting method not been available. In the Wasauksing First Nation survey, a similarly high percentage of respondents indicated that they would have voted anyway had online voting not been an option (88 percent), with 13 percent saying they ‘probably’ would not have. These modest percentages are consistent with studies of online voting adoption among Canadian municipal governments, which demonstrate internet voting can encourage a modest portion of less frequent voters to participate [6, 12].

Questions about voting histories were also asked, however, these were not consistent across surveys. Still, responses suggest that paper voters have more consistent voting records than those choosing to vote by internet. This may be because online voters are more likely to live off reserve and typically encounter a greater opportunity cost to attend a poll location (i.e. longer travel time). While these findings suggest positive implications for engagement, they should be taken with care given that these are self-reported measures.

Supporting the hypothesis that online voting has the potential to more consistently engage less frequent voters, paper voter respondents were asked how they would prefer to vote if unable to make it to a physical poll location. In Tsuut’ina Nation, 70 percent said they would vote by internet, 13 percent by mail, 6 percent by telephone, 4 percent would appoint a proxy to vote on their behalf and 2 percent would abstain from voting. In Wasauksing First Nation, by comparison, 49 percent said they would vote by internet, 22 percent by mail, 10 percent would appoint a proxy, 5 percent by telephone, 5 percent would abstain and 10 percent did not know. Of possible remote voting methods, online voting was by far the most desired way to cast a ballot in situations where electors could not make it to a polling location (i.e., being too busy, illness, inclement weather, transportation issues). The fact that a plurality of respondents in both First Nations selected online voting as their preferred voting method if unable to vote in person suggests it has the potential to enhance the participation of voters in special situations where otherwise attending a poll location may not be possible.

Taken together, survey findings indicate broad support for online voting. While paper voting is desired and is an important tradition to continue, online voting is a welcome addition to voting processes to enhance access, notably for citizens living off reserve and those residing on reserve who may encounter issues voting at a physical poll location.

6.4 Good Practices from Indigenous Experiences with Online Voting

Community-engaged work with First Nations and interviews with Indigenous leaders, practitioners and identified experts reveal a number of good practices for the implementation of online voting in Indigenous votes. Many of these are broad suggestions. However, this broad framing is suitable given that the adoption of online voting is highly contextual based on the circumstances surrounding its implementation and the unique features and needs of the community. Good practices can be grouped into four categories: (1) those relating to knowledge of the community, engagement, outreach and communication, (2) building tools and strategy, (3) clear processes and resources, and (4) a focus on technology.[2]

Broadly, under the theme of community knowledge and engagement good practices include: understanding community members – their demographics, preferences, and unique features which may make one type of voting model more successful than another – and what is needed for the particular vote (i.e., meeting a 25 percent quorum). Other good practices within this theme include robust education and communications about the online voting process, consultation with the community, and digital skill building, especially since this aspect was identified as an area where First Nations can be weak.

Second, building tools and strategies to support the successful deployment of online voting were communicated as good practices. This encompasses having accurate voters' lists and building email databases to reach as many members as possible. A third element involves recommending Indigenous communities take their time with online voting implementation, employing an incremental or iterative approach which involves a test and learn model wherein one or two things are tried at a time and then the approach is subsequently refined and expanded.

Third, having clear processes, resources and knowledge were identified as areas which could enhance online voting adoption. Part of this involves having a clear idea of who is in charge of the online voting aspect of Indigenous votes and outlining responsibilities of the First Nation and the technology vendor to minimize misunderstandings. Also, boosting technical knowledge is seen as critical to better enable election administrators to vet technology vendors and understand technical aspects of the vote to offer more secure voting options.

Finally, focusing on two key aspects of technology: security and access, are seen as good practices. Though no special ideas were suggested for how aspects of voting security may be unique in the context of Indigenous votes, each First Nation citizen has a status identification card number issued by the federal government. This is a unique identifier that could, and has been used as a credential to authenticate online voters. It was recommended by computer scientists that First Nations use a hybrid model of remote online voting whereby voters print a ballot they receive electronically, mark it and mail it back to election authorities. This approach maintains a paper record while improving access, albeit not to the same extent as with true remote online voting. In addition, ensuring access to online voting for those who want to use it and learn about it, but who may not otherwise have the resources to do so, is essential for promoting voter equality.

7 Discussion and Conclusion

Overall research conducted as part of the First Nations Digital Democracy project reveals interesting findings about online voting in Indigenous communities in Canada

[2] These are provided in a report prepared for Indigenous and Northern Affairs Canada with examples from Indigenous communities. To keep within length requirements, it was not possible to include a copy in this article.

and good practices that can be implemented to enhance deployment. Indigenous electors are willing to use online voting, voters are satisfied with it, and persons drawn to vote by paper are supportive of the policy change. Some paper voters are willing to try online voting in the future ‘no matter what’, whereas others (the largest proportion) say they would vote online in circumstances where they were unable to attend a polling location in person. Online voters are typically middle-aged, with slightly higher education and household income than paper voters. The youngest and oldest Indigenous voters are most inclined to vote by paper. In the case of younger voters we speculate this has to do with the fact that voting for the first time is a rite of passage and a more symbolic experience when carried out by paper, whereas for older voters concerns about access to the internet and electronic devices likely plays a large role in their willingness to not vote online.

One of the biggest takeaways from this research is that it appears online voting is an increasingly crucial tool to ensure community voice is incorporated in First Nation decision-making. This is especially true in cases where the federal government requires quorums be met for First Nations to pass their own laws and frameworks, and in instances where large portions of a community live off reserve. This is supported by our finding that online voters are more likely to live outside of the First Nation than paper voters, where travel to a polling location on reserve lands could make the voting prohibitively costly. Such findings are in line with other research which suggests voters living outside of their territories may have a disproportionate interest in voting online [11, 27].

In terms of potential improvements in engagement, further evidence is needed but our survey findings suggest that some of the people who chose online voting might not have participated otherwise. Past voting records of paper voters are also slightly more consistent than those of online voters. In this regard online voting could engage some of these electors on a more frequent basis given the greater convenience it offers. As noted, such findings are consistent with municipal studies of online voting in Canada [6, 12]. In addition, if paper voters were unable to attend a polling location, online voting would be their top choice.

Finally, our research presents a number of good practices for online voting implementation in Indigenous communities. While these lessons learned come from First Nations, much of their wisdom is transferrable to Indigenous communities in other countries as well as communities situated in rural and remote areas. Practices fall into four themed areas – those relating to knowledge of the community and outreach, building tools and taking an incremental approach, clear processes and increasing technical knowledge and resources, and addressing security and accessibility aspects of the technology.

Future studies could more deeply explore the effects of online voting on Indigenous participation, namely the degree to which it affects off reserve participation. As part of this studies could explore whether there are parallels between off reserve members’ uptake of online voting and use among expatriates in other countries such as Switzerland [11]. In addition, further examination of the challenges of inadequate internet access and lack of digital skills and how these areas can be improved is needed. Finally, comparative examinations of Indigenous deployment of online voting in other countries would be a welcome addition to the literature and our knowledge of how technology affects Indigenous peoples.

Acknowledgement. The authors would like to extend sincere thanks to Tsuut'ina Nation, Wasauksing First Nation, and Whitefish River First Nation as well as the many experts who participated in interviews. We acknowledge financial support from the Social Science and Humanities Research Council of Canada, the Government of Canada, and Chelsea Gabel's Canada Research Chair in Indigenous Well- Being, Community, Engagement and Innovation.

References

1. Alcantara, C., Dick, C.: Decolonization in a digital age: Cryptocurrencies and Indigenous self-determination in Canada. Canadian Journal of Law & Society/La Revue Canadienne Droit et Société. **32**,19-35 (2017).
2. Alvarez R.M., Katz, G., Pomares, J.:The impact of new technologies on voter confidence in Latin America: evidence from e-voting experiments in Argentina and Colombia. Journal of Information Technology & Politics. **8**, 199-217 (2011).
3. Bargh, M.: Opportunities and complexities for Māori and mana whenua representation in local government. Political Science. **68**, 143-160 (2016).
4. Belanger, Y.: You have to be involved to play a part in it: Assessing Kainai attitudes about voting Canadian elections. Great Plains Quarterly. **29**, 29-49 (2009).
5. Budd, B., Gabel, C., Goodman, N.: Technology in Indigenous communities in Canada: Implications for participation and governance. Conference Paper, Meeting of the Canadian Political Science Association, (2017).
6. Couture, J., Breux, S., Goodman, N.: La vote par Internet augmente-t-il la participation électorale? In: Loiseau H, Waldispuehl E (eds). Cyberespace et science politique: De la méthode au terrain, du virtuel au réel, pp 123-148. Presses de l'Université du Québec, Quebec City (2017).
7. Fleras, A., Elliott, J.L.: Unequal relations: An introduction to race, ethnic and Aboriginal dynamics in Canada. Prentice Hall, Scarborough (1999).
8. Gabel, C., Goodman, N., Bird, K., Budd, B.: Indigenous adoption of internet voting: A case study of Whitefish River First Nation. International Indigenous Policy Journal. 7, (2016).
9. Gabel, C., Goodman, N., Bird, K., Budd, B.: The impact of digital technology on First Nations participation and governance. The Canadian Journal of Native Studies **36**, 107-127 (2016).
10. Germann, M., Serdült, U.: Internet voting and turnout: Evidence from Switzerland. Electoral Studies **47,** 1-12 (2017).
11. Germann, M., Serdült, U.: Internet voting for expatriates: The Swiss case. JeDEM - EJournal of EDemocracy and Open Government **6,** 197–215 (2014).
12. Goodman, N.: Internet voting in a local election in Canada. In: Grofman B, Trechsel AH, Franklin, M (eds) The Internet and Democracy in Global Perspective. pp 7-24. Springer, Cham,(2014).
13. Goodman, N., Pammett, J.: The patchwork of internet voting in Canada. In: Electronic Voting: Verifying the Vote (EVOTE). pp 1-6. IEEE Press, New York, (2014).
14. Goodman, N., Pyman, H.: Understanding the effects of internet voting on elections: Results from the 2014 Ontario municipal elections. Technical Paper, Centre for e-Democracy, (2016).

15. Goodman, N., Stokes, L.C.: Reducing the Cost of Voting: An Evaluation of Internet Voting's Effect on Turnout. British Journal of Political Science. 1-13 (2018).
16. Green, G.: Participatory action research: Lessons learned with Aboriginal grandmothers. Health Care for Women International **22**, 471-482 (2001).
17. Israel, B., Schulz, A., Parker,, E., Becker, A.:. Review Of Community-Based Research: Assessing Partnership Approaches to Improve Public Health. Annual Review of Public Health. 19, 173-202 (1998).
18. Lockhart, E, Tenasco, A., Whiteduck, T., O'Donnell, S.: Information and communication technology for education in an Algonquin First Nation in Quebec. The Journal of Community Informatics **10**, (2013).
19. Maaka, R., Fleras, A.: The politics of Indigeneity: Challenging the state in Canada and Aotearoa New Zealand. Otago University Press, Dunedin (2006).
20. McMahon, R., Gurstein, M., Beaton, B., O'Donnell, S., Whiteduck, T.: Making information technologies work at the end of the road. Journal of Information Policy **4**, 250-269 (2014).
21. McMahon, R.: Creating an enabling environment for digital self-determination. Media Development **2**, 11-15 (2014).
22. Midzain-Gobin, L., Goodman, N., Gabel, C., Bird, K.: Time for change? Reforming the Indian Act to allow for online voting. Policy Options Special Issue: The Indian Act: Breaking its Stubborn Grip (2017).
23. Monchalin, L.: The colonial problem: An Indigenous perspective on crime and injustice in Canada. University of Toronto Press, Toronto (2016).
24. Norris, P.: Will new technology boost turnout? Evaluating experiments in UK local elections. In Kersting N, Baldersheim H (eds) Electronic voting and democracy. pp 193-225. Palgrave Macmillan, London, (2004).
25. O'Donnell, S., Beaton, B., McMahon, R., Hudson, H., Williams, D., Whiteduck, T., First Nations Education Council: Digital technology adoption in remote and northern Indigenous communities in Canada. Conference Paper, Meeting of the Canadian Sociological Association, (2016).
26. Segaard, S., Baldersheim, H., Saglie, J.:The Norwegian trial with internet voting: results and challenges. In: Barrat J, Esteve I (eds) El voto electrónico y sus dimensiones jurídicas: entre la ingenua complacencia y el rechazo precipitado. Iustel, Madrid (2016).
27. Serdült, U., Germann, M., Mendez, F., Portenier, A., Wellig, C.: Fifteen years of internet voting in Switzerland: History, governance and use. In: 2015 2nd International Conference on eDemocracy and eGovernment, pp 126–132, IEEE Press, New York, (2015).
28. Smith, L.: Decolonizing methodologies: Research and indigenous peoples. Zed Books Ltd, London (2013).
29. Solop, F.: Digital Democracy Comes of Age: Internet Voting and the 2000 Arizona Democratic Primary Election. Political Science & Politics. 34, 289-293 (2001).
30. Trechsel, A., Vassil, K.: Internet voting in Estonia. A comparative analysis of four elections since 2005. Technical Report, Council of Europe, (2010).
31. Truth and Reconciliation Commission of Canada: Calls to Action. http://www.trc.ca/websites/trcinstitution/File/2015/Findings/Calls_to_Action_English2.pdf (2015).
32. Vassil, K., Solvak, M., Vinkel, P., Trechsel, A., Alvarez, A.:The diffusion of internet voting. Usage patterns of internet voting in Estonia between 2005 and 2015. Government Information Quarterly **33**, 453-459 (2016)

Internet Voting User Rates and Trust in Switzerland

Uwe Serdült[1, 2] [0000-0002-2383-3158] and Victor Kryssanov[1]

[1] College of Information Science and Engineering, Ritsumeikan University, Kusatsu, Japan
[2] Center for Democracy Studies Aarau (ZDA) at the University of Zurich, Aarau, Switzerland

serdult@fc.ritsumei.ac.jp

When remote internet voting is available as a voting channel for Swiss citizens living in Switzerland, user rates are surprisingly low when compared to general internet penetration rates. Switzerland is a highly decentralized polity. Cantons are thus free to choose which voting channels and which internet voting models they want to offer. Internet voting adoption patterns therefore depend on the model a canton opted to implement: a) with pre-registration user rates started within single digits and took more than a decade to grow above 10 percent; b) without we observed initial spikes above 30 percent (due to the novelty effect) followed by a drop and stagnation around a user rate of 20 percent. Internet voting user rates are on a satisfactory level of 60 percent only for Swiss living abroad [1] because for them the alternative is postal voting with a risk for delays. However, the Swiss domestic trajectories in principle also hold for cases elsewhere [2, 4]. If one wants to see the glass as half full we could say that time itself will fix the problem. Younger cohorts of voters and increasing digitalization will push user rates up in general. However, even in Estonia where internet voting is by now well established, user rates tend to be saturating on a level only slightly above the 30 percent level. In sum, internet voting user rates are in general lower than we would expect them to be.

Fig. 1. Support for trust building measures among Swiss citizens (*n* = 1228)

Proposition	Trust increase	No trust increase	Don't know/NA
Testing internet voting on a demo website	63	28	9
Making the source code public	22	21	57
Repeated voting until election day	22	68	10
Verifiable vote with code on the ballot	68	22	11
Security audits by external experts	55	33	12

Besides the convenience and popularity of generalized postal voting in Switzerland, the lack of trust in Swiss internet voting seems to be one of the main impediments for higher user rates. Looking at recent survey data [3], the mean trust score on a scale from 0 to 10 for voting at the polling station is 8.5, for mail-in ballots 8.2 and 6.6 for internet voting.

As we can see in Table 1, the potential to increase trust in internet voting among the public in the future seems to be limited since most measures are either already implemented (audits, demo website, verification code), are not publicly addressable (source code), or do not fit Swiss political culture (repeated voting).

We therefore propose to explore unconventional avenues. Based on social identity theory, [5] suggested that the acceptance of internet voting increases if the people administering the solution are perceived as being of their own. In the Swiss case, this would mean that internet voting would need to be operated in a decentralized manner by each local election management board separately. For ballot and postal voting this is the norm. Despite of the logistical challenge one should look into the decentral operation of internet voting as well. In addition and paradoxically, in our view, the re-materialization of internet voting could provide a solution for this conundrum as well. If a secure decentralized network would be used to “materialize” individual votes and the whole voting process in the form of either an append only “paper trail” or 3D shapes resistant to replicating and reverse-engineering, the requirement for irreplaceable tangible, re-countable objects would be fulfilled. Prototypes of internet voting distributed ballot boxes should therefore be developed and tested.

References

1. Germann, M., Serdült, U. Internet voting for expatriates: The Swiss case. JeDEM-eJournal of eDemocracy and Open Government 6(2), 197-215 (2014).
2. Goodman, N., Smith, R. Internet Voting in Sub-national Elections: Policy Learning in Canada and Australia. In: Krimmer, R., Volkamer, M., Barrat, J. et.al. Electronic Voting: First International Joint Conference, E-Vote-ID 2016, Bregenz, Austria, October 18-21, 2016, Proceedings, Lecture Notes in Computer Science, pp. 18-21. Springer, Berlin (2017).
3. Milic, T., McArdle, M., Serdült, U.: Attitudes of Swiss citizens towards the generalization of e-voting. ZDA, Aarau (2016). https://doi.org/10.5167/uzh-127938
4. Saglie, J., Bock Segaard, S. Internet voting and the secret ballot in Norway: principles and popular understandings. Journal of Elections, Public Opinion and Parties 26(2), 155-169 (2016).
5. Warkentin, M., Sharma, S., Gefen, D., Rose, G. M., Pavlou, P. Social identity and trust in internet-based voting adoption. Government Information Quarterly 35(2), 195-209 (2018).

Counting Functions | Blockchain

Rounding Considered Harmful

Carsten Schürmann[*]

DemTech
IT University of Copenhagen
Denmark
carsten@itu.dk

Abstract. Party-list proportional representation methods aim to allocate seats proportionally to the votes cast for each party. In general, exact proportionality is not possible as it would require a fractional allocation of seats. Therefore several methods have been devised to compute seat allocations that differ in the way they try to achieve proportionality. Examples of such methods include the d'Hondt and Sainte-Laguë methods that allocate seats according to fractional values. These methods are used in many countries. Numerically, these fractions appear harmless, however they are not. Computers do not work with infinite precision floating point numbers, implementations tend to round the fractions to several digits, which can, with a certain probability, lead to incorrect seat allocations.

1 Introduction

Denmark's electoral law requires a combination of d'Hondt and Sainte-Laguë methods to compute the seat allocation of parliament after a parliamentary election. These methods aim at producing a seat allocation that is proportional and reflects the vote. The methods are deceptively simple as they consist of computing a table of quotients and the selection of the largest those quotients, each of which corresponds to a seat.

However, whenever quotients are computed, one has to be careful with bad numeric effects. Floating point numbers are not represented with infinite precision, and often the digits after the comma are rounded off. In this paper we ask the question if it safe to round quotients, and if rounding can have an effect in real elections. The answer is that it is not safe to round and that rounding can change the outcome of the election (with a certain probability).

For conducting this work, we have been granted access the source code of Denmark's Seat allocation System (DSAS). While inspecting the software, we observed that DSAS rounds quotients to the next whole number before storing

[*] This paper was made possible by grant NPRP 097-988-1-178, Automated verification of properties of concurrent, distributed and parallel specifications with applications to computer security, from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the author.

then in the table, and then uses randomness to draw lots to break ties. In the analysis described in this paper, we show that this is highly problematic and that situation can arise where DSAS computes the wrong result. We also conduct a Monte-Carlo experiment to provide some statistical evidence, how likely the error scenario actually is in practice, at least for the Danish case. We estimate that one out of every 66 Danish elections is effected and that DSAS computes the wrong seat allocation in average for one out of 132 elections. The version of DSAS under consideration in this paper was used only since 2007. Seat allocation for elections before 2007 were computed with another system. We have not conducted any further statistical analysis for other countries.

We have written this paper out of concern that similar programming mistakes may be hiding in seat allocations programs used by other countries.

This paper is organized as follows. In Section 2 we describe abstractly the d'Hondt method, variants of which are used in dozens countries around the world. In Section 3, we describe the effects of rounding and define the probability of how prematurely rounding quotients can effect the result of the seat allocation algorithm. In Section 4 we then look at real election data from the Danish 2007 parliament election, and argue, that the effects of rounding are likely and serious. We also provide evidence that the Danish Seat Allocation System, at least in the version examined, rounds quotients to the next whole number and breaks ties by drawing lots. In Section 5 we describe briefly how we disclosed the findings to the Ministry, before we we assess results and describe how a fix for the rounding problem in Section 6.

Related Work: The subject of this paper touches on several different areas: Party-list proportional representation methods are social choice functions, and their properties have been studied in political science and social choice [1, 5]. The d'Hondt was originally invented by Thomas Jefferson in 1791, and then introduced to Europe by Victor d'Hondt in 1878. The method was designed to be executed by hand. Nowadays, social choice functions such as d'Hondt, single transferable vote (STV) and others are implemented as computer programs. Programming mistakes are common place unless one uses formal methods to verify the implementation of tie breaking rules, which not many implementors do. One of the few works in this area is by Goré and Lebedeva [4], which focuses on verifying implementations of STV and the respective tie-breaking rules. Their reasoning techniques applied to d'Hondt (assuming it is correctly specified) would then automatically recognize programming mistakes such as those that we discuss in this paper. Lastly, even if premature rounding is used in the implementation of d'Hondt and Sainte-Laguë methods, variants of risk-limiting auditing tailored for d'Hondt elections [7] can be used to identify statistically, if seats have been erroneously assigned to the wrong party.

2 The d'Hondt Method

The d'Hondt and Sainte-Laguë methods are a part-list proportional representation methods used for seat allocation in more than 50 countries including

Denmark, Germany, and Switzerland. In the following, we focus on d'Hondt. It is defined as follows. Let $t_1 \dots t_n$ represent the vote totals of an election with n parties. The total number of votes cast is therefore $\sum_{1\leq i\leq n} t_i$, and the goal of the d'Hondt method is to assign the m seats in such a way that they are proportionally allocated in the the number of votes obtained by each party: As there are no fractional seats, we cannot expect d'Hondt or any other voting rule for this matter to produce the perfectly proportional seat allocation. The underlying idea of d'Hondt is this: If a party were to pay for a seat with votes then then d'Hondt rule allocates the number of seats to each party to maximize the highest (average) price per seat.

Example 1. Let A and B be two parties with 10000 and 15000 votes, respectively, and five seats to be allocated.

A's bid	B's bid	Allocation
(1) $10,000$ for 1 seat	$15,000$ for 1 seat	B
(2) $10,000$ for 1 seat	$15,000$ for 1 seat	B, A
(3) $10,000$ for 1 seat	$7,500$ for 2 seats	B, A, B
(4) $5,000$ for 2 seats	$7,500$ for 2 seats	B, A, B, A
(5) $5,000$ for 2 seats	$5,000$ for 3 seats	B, A, B, A, B

The algorithm starts with highest price, here $15,000$ votes, and reduces the price until all seats are sold! In line (1), the first seat goes to party B, because B is bidding $15,000$. In line (2), A obtains the second seat, because at this stage the price is $10,000$. B cannot bid, because B has spent all of its money. In line (3), the average price for a seat has dropped to $7,500$, which allows B to argue that it should be entitled to a second seat. (B has already spent $15,000$ and $2 \times 7,500 = 15,000$). After 5 rounds, all seats are sold.

The d'Hondt rule results in a simple algorithm that consists of two steps.

1. Construct a table, one column per party, where the first row are initialized with the vote totals $t_1 \dots t_n$. All other rows are identified by a divisor, and the row is computed from the first row by dividing t_i by this divisor. The entries in the table are also called *quotients*. In the simplest case, the divisors range over $1, 2, 3, \dots$, but other choices of divisors are used in practice as well, as we will see in Section 4.
2. To allocate s seats, traverse the table and mark the s highest quotients. The number of markings in each column correspond to the seats assigned for the respective party.

The intended meaning of the table is that the field located at row i and row j is the bidding price for party i for j seats. Note, that if a quotient is marked in a table, all the quotients above (in the same column) are also marked.

Example 2. Back to the example above. The table in this case has the following form. The markings are indicated as check marks.

Divisor	Party A	Party B
1	10000 ✓	15000 ✓
2	5000 ✓	7500 ✓
3	$3333.\bar{3}$	5000 ✓
4	2500	3750
5	2000	3000

3 To Round or Not to Round?

Although computing the table is not difficult there are some decision designs that have to be taken when implementing it. The most important perhaps is if and how to round quotients that are not whole numbers. The quotient in Row 3, Party A, for example, is a number with infinitely many digits after the comma. Would it be ok to round these quotients to the nearest whole number? By rounding to the next whole number we mean that if the digits after the "," < 0.5 the number will be rounded down and if ≥ 0.5 it will be rounded up. One may expect the answer is yes, after all, the differences between the tallies of a typical national election are usually quite large, so what could go wrong? Rounding must be considered problematic, if it affects the result of seat allocation, which means that the margin between two quotients in the d'Hondt table are sufficiently close, and this means introducing a tie that has to be resolved by a tie-breaking rule.

Example 3. Consider, for example, a multi-member constituency where three parties A, B and C with tallies 999, 500, 1501, respectively, compete for four constituency seats. Selecting the four highest quotients from the table below

	A	B	C
Tallies	999	500	1501
Divisor 1	999	500	1501
Divisor 2	499.5	250	750.5
Divisor 3	333	$166.\bar{6}$	$500.\bar{3}$

results in the following correct election result: one seat is allocated to A, three seats are allocated to C. In the case with rounding to the next whole number, the table has the following form:

	A	B	C
Tallies	999	500	1501
Divisor 1	999	500	1501
Divisor 2	500	250	750
Divisor 3	333	167	500

The algorithm allocates the first three seats to C, A, and C, and the last seat is drawn by lot, and as each party has a quotient of 500, the probability that the correct result is drawn is only 1/3.

If we were to allocate five seats instead of four, the correct result implies that the fifth seat belongs to B, because $500 > 499.5$. In the rounded version, however, we would have to draw two seats of the set of three, which means here again, the probability that the correct result is only $1/3$.

In real elections there are several factors that impact the margins of the quotient registered in the table: (1) In some countries, d'Hondt is applied not only to the nation-wide totals, but often also on the constituency level, where the tallies are much smaller, which means the likelihood that two quotient are close increases. (2) The choice of divisors varies form country to country. In some countries, for example in Denmark, under certain but rare circumstances, the divisors $1, 4, 7 \ldots$ are being used to construct the d'Hondt table, which means that for large divisors, the quotients, can become quiet close. (3) The size of the elected body plays a big role in how small the margins are between any to quotients in the table.

We distinguish to kinds of errors due to rounding, depending on if they affect allocated seats alone, or allocated and non-allocated seats. The former is called *Incorrect Allocation Order* is relatively harmless, because it does not affect the overall election result, but only the evidence of how the election result was determined. In some countries already this might be considered an infringement of the law. Things become much more worrisome, when rounding creates an artificial tie situation affecting the last seat(s) to be awarded. In this case, the drawing of the lots may in fact the overall election result. This error is called *Incorrect Seat Allocation.*

To make our analysis precise, we need to distinguish between two n-way tie situations: We say an n-way tie between n quotients is *genuine*, if the quotients (before rounding) are all equal. In the case that the quotients (before rounding) are not equal, but they are equal after rounding, we speak of a *false* n-way tie situation. A genuine tie must be broken by drawing lots, whereas a false tie must not. For the following two definitions, we assume a false tie situation.

Error 1: Incorrect Allocation Order In the case that the number of seats to be allocated exceeds the number of k quotients rounded to the same number, drawing lots may affect the order in which the seats are allocated, but it does not effect the overall election result. For a false tie, the probability p of allocating seats in the wrong order is

$$p = \frac{1}{k!}.$$

Error 2: Incorrect Seat Allocation In the case that the number of quotients rounded to the same number exceeds the number of seats to be allocated, lots will have to be drawn, which means that seats may be allocated in error with non-negligible probability. More precisely, if there is a false tie between n rounded quotients and only $k(< n)$ seats are left to be allocated, the probability p that the result of seat allocation is correct result is precisely

$$p = \frac{1}{\binom{n}{k}}.$$

4 Case Study Denmark

In order to learn if this a real or just an academic problem, we describe a case study that we conducted in Denmark. We discuss the Danish legal framework in Section 4.1, empirical evidence that Errors 1 and 2 could have actually been encountered in Section 4.2, and a discussion of the implementation of Denmark's seat allocation system, in Section 4.3 where we demonstrate that the system actually rounds quotients.

4.1 Legal Framework

Danish election law[1] defines the rules for how mandates are to be distributed. The law distinguishes two kinds of seats, constituency seats *kredsmandater* and compensatory seats *tillægsmandater*. In this report, we focus mostly only compensatory seats, but our findings also apply to the calculation of constituency seats. We quote the relevant sections from the Danish Parliamentary Elections Act [3].

Allocation of Constituency Seats

> §76. (1) The votes cast for each party in all nomination districts in a multi-member constituency shall be summed up. The votes cast for each individual candidate shall equally be summed up.
>
> (2) Each number of votes appearing as a result of the summation, cf. subsection (1), shall be divided by 1 - 2 - 3 and so on until such number of divisions equivalent to the maximum number of seats expected to be allocated to the party or to the independent candidate has been performed. The party or the independent candidate having the highest resulting quotients shall be given the first seat in the multi-member constituency. The second highest quotient entails the second seat and so on and so forth, until all constituency seats in the multi-member constituency have been distributed among the parties and the independent candidates. If two or more quotients are of equal size, lots shall be drawn.

Allocation of Compensatory Seats to Parties by Region

> §78.(1) For each of the parties which are allocated compensatory seats according to section 77, the number of votes cast for the party in each of the three regions shall be computed.
>
> (2) Each of these votes shall be divided by the figures 1 - 3 - 5 - 7 and so on. Next, a number of the largest quotients equivalent to the number of constituency seats obtained by the party in the region according to section 76 shall be omitted.

[1] LBK nr 416 af 12/05/2016

> (3) The region and the party which subsequently has the largest quotient shall have the first compensatory seat. The region and the party which has the second largest quotient shall have the next compensatory seat and so on and so forth. Where a region or a party has obtained the number of compensatory seats it should have, cf. sections 10 and 77, the region or the party shall not be considered any further. The allocation continues for the other regions and the other parties until all compensatory seats have been distributed. If a party which has not received votes in all three regions cannot be allocated the compensatory seats to which the party is entitled by this distribution, these seats shall be allocated in advance to the party in the regions where votes have been cast in its favor.

This law text describes the social choice function to compute the seat allocation for the 135 constituency seats and 40 compensatory seats in the Danish Parliament, called *Folketinget*. We focus our attention on the allocation of compensatory seats, because the divisors become larger and the quotients smaller than those when allocating constituency seats. Denmark is divided into three regions, *Hovedstaden*, *Sjælland-Syddanmark*, and *Midtjylland-Nordjylland*.

The technique described in the law is a variant of d'Hondt, as described in Section 2 (see §10 (2), *den største brøks metode*). Constituency seats are assigned using divisors $1, 2, 3, \ldots$, and compensatory seats are allocated in a second step using divisors $1, 3, 5 \ldots$ In the case of a tie, lots shall be drawn. We want to emphasize, that neither the published versions of the largest remainder methods, nor the Danish election law permits that rounded versions of the quotients may be considered. This interpretations has been confirmed by the Danish Ministry.

4.2 Empirical Analysis

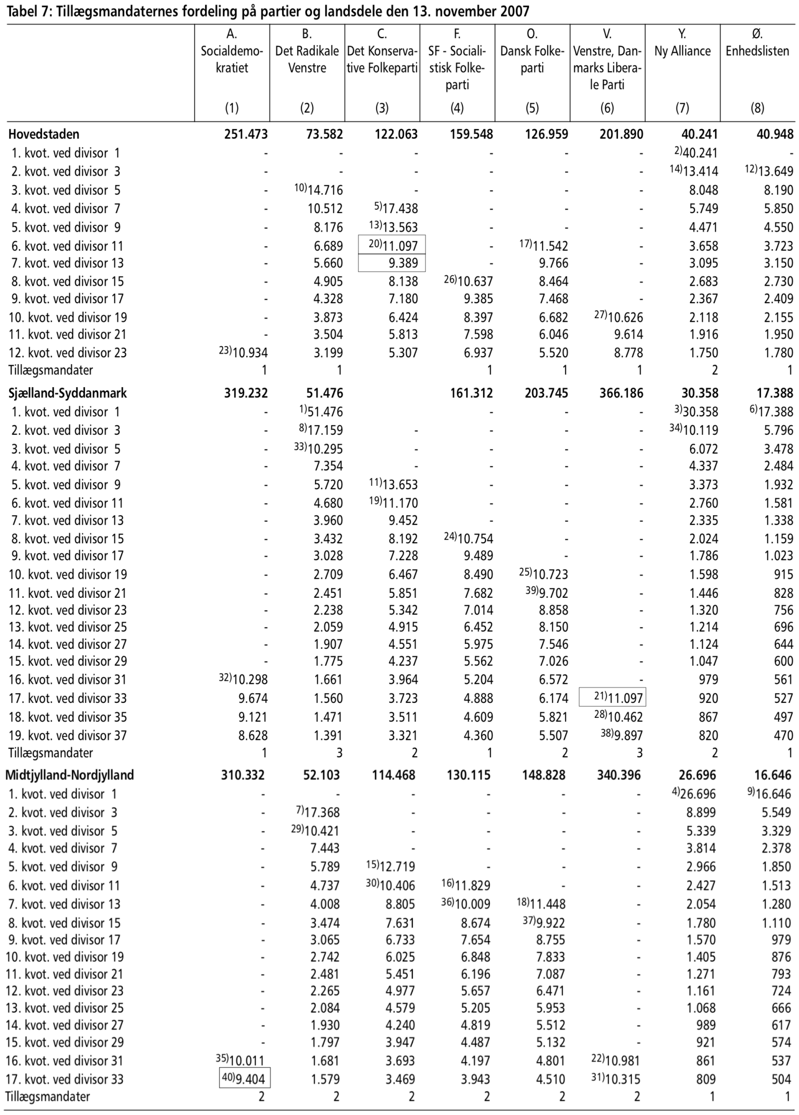
For our empirical analysis, the interesting step is 3. where the quotients are computed. Consider the seat allocation of compensatory seats 20 and 21 during Denmark's parliamentary election in 2007, as depicted in Figure 1, which has been taken from [2, page 17]. This election is relatively recent and it was chosen, because it demonstrates that Error 1 and 2 do actually arise in real elections. Both seats were awarded with a rounded quotient of $11,097$. The seat 20 was awarded to *Det Konservative Folkeparti*, because

$$\frac{122063}{11} = 11,096.6363636$$

and seat 21 was awarded to *Venstre* because

$$\frac{366186}{33} = 11,096.5454545.$$

In Figure 1, we have marked the important entries using boxes to help the reader identify these more easily.



Tabel 7: Tillægsmandaternes fordeling på partier og landsdele den 13. november 2007

	A. Socialdemokratiet (1)	B. Det Radikale Venstre (2)	C. Det Konservative Folkeparti (3)	F. SF - Socialistisk Folkeparti (4)	O. Dansk Folkeparti (5)	V. Venstre, Danmarks Liberale Parti (6)	Y. Ny Alliance (7)	Ø. Enhedslisten (8)
Hovedstaden	**251.473**	**73.582**	**122.063**	**159.548**	**126.959**	**201.890**	**40.241**	**40.948**
1. kvot. ved divisor 1	-	-	-	-	-	-	$^{2)}$40.241	-
2. kvot. ved divisor 3	-	-	-	-	-	-	$^{14)}$13.414	$^{12)}$13.649
3. kvot. ved divisor 5	-	$^{10)}$14.716	-	-	-	-	8.048	8.190
4. kvot. ved divisor 7	-	10.512	$^{5)}$17.438	-	-	-	5.749	5.850
5. kvot. ved divisor 9	-	8.176	$^{13)}$13.563	-	-	-	4.471	4.550
6. kvot. ved divisor 11	-	6.689	$^{20)}$11.097	-	$^{17)}$11.542	-	3.658	3.723
7. kvot. ved divisor 13	-	5.660	9.389	-	9.766	-	3.095	3.150
8. kvot. ved divisor 15	-	4.905	8.138	$^{26)}$10.637	8.464	-	2.683	2.730
9. kvot. ved divisor 17	-	4.328	7.180	9.385	7.468	-	2.367	2.409
10. kvot. ved divisor 19	-	3.873	6.424	8.397	6.682	$^{27)}$10.626	2.118	2.155
11. kvot. ved divisor 21	-	3.504	5.813	7.598	6.046	9.614	1.916	1.950
12. kvot. ved divisor 23	$^{23)}$10.934	3.199	5.307	6.937	5.520	8.778	1.750	1.780
Tillægsmandater	1	1		1	1	1	2	1
Sjælland-Syddanmark	**319.232**	**51.476**		**161.312**	**203.745**	**366.186**	**30.358**	**17.388**
1. kvot. ved divisor 1	-	$^{1)}$51.476		-	-	-	$^{3)}$30.358	$^{6)}$17.388
2. kvot. ved divisor 3	-	$^{8)}$17.159	-	-	-	-	$^{34)}$10.119	5.796
3. kvot. ved divisor 5	-	$^{33)}$10.295	-	-	-	-	6.072	3.478
4. kvot. ved divisor 7	-	7.354	-	-	-	-	4.337	2.484
5. kvot. ved divisor 9	-	5.720	$^{11)}$13.653	-	-	-	3.373	1.932
6. kvot. ved divisor 11	-	4.680	$^{19)}$11.170	-	-	-	2.760	1.581
7. kvot. ved divisor 13	-	3.960	9.452	-	-	-	2.335	1.338
8. kvot. ved divisor 15	-	3.432	8.192	$^{24)}$10.754	-	-	2.024	1.159
9. kvot. ved divisor 17	-	3.028	7.228	9.489	-	-	1.786	1.023
10. kvot. ved divisor 19	-	2.709	6.467	8.490	$^{25)}$10.723	-	1.598	915
11. kvot. ved divisor 21	-	2.451	5.851	7.682	$^{39)}$9.702	-	1.446	828
12. kvot. ved divisor 23	-	2.238	5.342	7.014	8.858	-	1.320	756
13. kvot. ved divisor 25	-	2.059	4.915	6.452	8.150	-	1.214	696
14. kvot. ved divisor 27	-	1.907	4.551	5.975	7.546	-	1.124	644
15. kvot. ved divisor 29	-	1.775	4.237	5.562	7.026	-	1.047	600
16. kvot. ved divisor 31	$^{32)}$10.298	1.661	3.964	5.204	6.572	-	979	561
17. kvot. ved divisor 33	9.674	1.560	3.723	4.888	6.174	$^{21)}$11.097	920	527
18. kvot. ved divisor 35	9.121	1.471	3.511	4.609	5.821	$^{28)}$10.462	867	497
19. kvot. ved divisor 37	8.628	1.391	3.321	4.360	5.507	$^{38)}$9.897	820	470
Tillægsmandater	1	3	2	1	2	3	2	1
Midtjylland-Nordjylland	**310.332**	**52.103**	**114.468**	**130.115**	**148.828**	**340.396**	**26.696**	**16.646**
1. kvot. ved divisor 1	-	-	-	-	-	-	$^{4)}$26.696	$^{9)}$16.646
2. kvot. ved divisor 3	-	$^{7)}$17.368	-	-	-	-	8.899	5.549
3. kvot. ved divisor 5	-	$^{29)}$10.421	-	-	-	-	5.339	3.329
4. kvot. ved divisor 7	-	7.443	-	-	-	-	3.814	2.378
5. kvot. ved divisor 9	-	5.789	$^{15)}$12.719	-	-	-	2.966	1.850
6. kvot. ved divisor 11	-	4.737	$^{30)}$10.406	$^{16)}$11.829	-	-	2.427	1.513
7. kvot. ved divisor 13	-	4.008	8.805	$^{36)}$10.009	$^{18)}$11.448	-	2.054	1.280
8. kvot. ved divisor 15	-	3.474	7.631	8.674	$^{37)}$9.922	-	1.780	1.110
9. kvot. ved divisor 17	-	3.065	6.733	7.654	8.755	-	1.570	979
10. kvot. ved divisor 19	-	2.742	6.025	6.848	7.833	-	1.405	876
11. kvot. ved divisor 21	-	2.481	5.451	6.196	7.087	-	1.271	793
12. kvot. ved divisor 23	-	2.265	4.977	5.657	6.471	-	1.161	724
13. kvot. ved divisor 25	-	2.084	4.579	5.205	5.953	-	1.068	666
14. kvot. ved divisor 27	-	1.930	4.240	4.819	5.512	-	989	617
15. kvot. ved divisor 29	-	1.797	3.947	4.487	5.132	-	921	574
16. kvot. ved divisor 31	$^{35)}$10.011	1.681	3.693	4.197	4.801	$^{22)}$10.981	861	537
17. kvot. ved divisor 33	$^{40)}$9.404	1.579	3.469	3.943	4.510	$^{31)}$10.315	809	504
Tillægsmandater	2	2	2	2	2	2	1	1

Fig. 1. Compensatory seat allocation.

If we were to use instead a seat allocation algorithm that rounds all quotients to the nearest whole number, these two quotients would round to $11,097$ and consequently seat 20 and seat 21 can only be allocated by drawing lots. In this situation, the following two outcomes are equally likely.

1. Seat 20 will be allocated to *Det Konservative Folkeparti* and seat 21 to *Venstre*.

2. Seat 20 will be allocated to *Venstre* and seat 21 to *Det Konservative Folkeparti*.

Both outcomes are correct, as we have discussed in Section 2, however Denmark's law does regard — strictly speaking — only the first outcome as correct and the second outcome as an instance of Error 1. Evidently, the order in which the seats were assigned was correct, perhaps because it was computed with an earlier version of seat allocation system and not the one we will discuss below.

Regarding Error 2, we notice that the last compensatory seat (Seat 40) was awarded with a quotient of exactly 9404 to the *Socialdemokratiet*. The next highest quotient in Figure 1 is is $9,389$ but what cannot be seen in the table is that there are actually two entries: One for region *Hovedstaden*, *Det Konservative Folkeparti*, divisor 13,

$$\frac{122063}{13} = 9389.46153846$$

and the other for region *Sjælland-Syddanmark*, *Venstre*, divisor 39

$$\frac{366186}{39} = 9389.38461538.$$

This means that in the hypothetical situation where the Danish Parliament had 41 compensatory seats, a rounding seat allocation system might have assigned the 41^{st} seat to the wrong party with a probability of 50%, i.e. to *Venstre* instead of *Det Konservative Folkeparti*.

The remaining question is, of course, how big of a problem Error 1 and Error 2 in practice really are. A statistical analysis proves difficult, in part because of the many random variables that need to be considered. Therefore we resort to a Monte-Carlo experiment and develop an election simulator to compute the probabilities of false n-way ties. In our experiment, where we work with 8 parties with tallies chosen at random between $20,000$ and $400,000$ votes, a situation, which pretty accurately describes the parameters of a Danish election. We then run the simulator $1,000,000$ times where we compute a d'Hondt table with 50 rows (which corresponds to a highest divisor of 101).

Error 1: The following table depicts the expected value of n-way ties occurring in a single d'Hondt table.

$n =$	1	2	3	4
E(false n-way tie)	173.17545	0.904989	0.0048	0.000022

This means that in 9 of of 10 d'Hondt tables, there will be in average one false 2-way tie that is decided by drawing lots. However, our experiment also shows that the risk of an Error 1 in the cases of false 3- and 4-way ties are extremely rare. Recall, that Error 1 will not change the election result but only the order in which seats are assigned.

Error 2: In the same statistical experiment, we look at the 175th seat being chosen. The following table depicts the number of false n-way ties observed.

$n =$	1	2	3	4
# observations	$984,089$	$15,766$	144	1

This means, that the probability of a false 2-way tie situation is roughly 1.5%, which means that in one out of 66 elections, the last seat will be awarded with drawing lots.

In summary, rounding while computing the d'Hondt and Sainte-Laguë tables should be considered harmful. We have demonstrated that there is a non-negligible chance that seats are being allocated in the wrong order and/or to the wrong party.

4.3 Implementation

The software for Denmark's Seat Allocation System (DSAS) has been implemented by Statistics Denmark (*Danmarks Statistik*), in a programming language called PL/SQL (part of the official ORACLE database distribution) used to program stored procedures. DSAS implements the problematic rounding version of the seat allocation algorithm outlined in the previous section. In a nutshell, the implementation rounds to the nearest number and then generate randomly the digits after the comma as a tie breaker. For the above example, instead of computing the quotients precisely, DSAS rounds and randomly generates after comma digits.

Fraction	Correct result	DSAS result
$\frac{122063}{11}$	$11,096.6363636$	$11,096 + r_1$
$\frac{366186}{33}$	$11,096.5454545$	$11,096 + r_2$

where $r_1, r_2 \in [0, 1)$ randomly chosen.

The problematic code can be found in the file `packages.sql`, dated 16. September 2013, 16:12. The three procedures that support our claim regarding the allocation of compensatory seats are depicted in Figure 2 that describes how quotients for compensatory seats are computed, Figure 3 that illustrates how the quotients are introduced into the main table, and Figure 4 that demonstrates how the quotients are ordered for further computation. The program for allocating the constituency seats is very similar, suffers from the same rounding problem, and can be found elsewhere in this file.

Tillaeg_landsdel_dankvot_ins In Figure 2, the loop starting in line 23158 ranges over all possible divisors, starting from 1 until the maximal number stored in variable `V_antal_divisioner`. The body of the loop consists of two steps. In the first step, a temporary table `TempTab` is defined that stores all possible quotients (in no particular order). In lines 23165–23174 it is determined what precisely is stored in `TempTab`. The two critical lines here are 23172 and 23173. In the former DSAS uses SQL's rounding function to compute `round (Antal_stemmer / V_divisor)`, which computes the quotient rounded to the nearest whole number. The string `Kvotient` tells the ORACLE database engine to name the column `Kvotient`. In the latter DSAS stores a random value using Oracle's random generator in a column called `Random_nr` as a tie breaker.

```
for I in 1 .. V_antal_divisioner loop
  if I = 1 then
    V_divisor := 1;
  else
    V_divisor := V_divisor + 2;
  end if;

  select Obj_tillaegsmandater_landsdele
              (Landsdel_id, Parti_id, Antal_stemmer,
               Antal_kredsmandater, Kvotient_nr, Kvotient, Random_nr)
    bulk collect into Temptab
    from (select Landsdel_id,
                 Parti_id,
                 Antal_stemmer,
                 Antal_kredsmandater,
                 V_divisor Kvotient_nr,
                 round (Antal_stemmer / V_divisor) Kvotient,
                 dbms_random.Normal Random_nr
            from table (P_col_tillaegsman_landsdele));

  Insert_tillaeg_landsdel_kvot (Temptab,
                                P_log_bruger_in,
                                P_term_bruger_in,
                                P_valg_id_in,
                                P_valgfase_in,
                                P_debug_in,
                                P_log_id_in);
end loop;
```

Fig. 2. Procedure `Tillaeg_landsdel_dankvot_ins`

How precisely Oracle's random generator was seeded, could not be determined. As we will see below, if two quotients are compared and if the Kvotient part is equal, the Random_nr will determine which of the two is ranked higher. In the second step, after computing all quotients, DSAS calls a function to copy the quotients from TempTab into the right table using a stored procedure called Insert_tillaeg_landsdel_kvot (see line 23176), which we discuss next. Note that the first argument to this method is TempTab.

```
23384  procedure Insert_tillaeg_landsdel_kvot (
...
23404    insert into Tillaeg_man_land_kvotienter (Landsdel_id,
23405                                              Parti_id,
23406                                              Antal_kredsmandater,
23407                                              Tmk_land_kvotientnr,
23408                                              Tmk_land_kvotient,
23409                                              Tmk_random_nr,
23410                                              Valgfase_kode,
23411                                              Tmk_koersel_id,
23412                                              Valg_id)
23413      select Landsdel_id,
23414             Parti_id,
23415             Antal_kredsmandater,
23416             Kvotient_nr,
23417             Kvotient,
23418             Random_nr,
23419             P_valgfase,
23420             P_log_id_in,
23421             P_valg_id_in
23422        from table (P_col_tillaegsman_landsdele);
```

Fig. 3. Procedure Insert_tillaeg_landsdel_kvot

Insert_tillaeg_landsdel_kvot Figure 3 depicts a procedure that simply reads all tuples from the table referred to by first argument, i.e. TempTab. The destination table is Tillaeg_man_land_kvotienter, where the attributes for the table are renamed to Tmk_land_kvotient (line 23408) and Tmk_random_nr (line 23409), respectively.

The tuples in the table Tillaeg_man_land_kvotienter are stored in no particular order.

Tillaeg_landsdel_hentpotkvot A fragment of the procedure that accesses and sorts the table Tillaeg_man_land_kvotienter is depicted in Figure 4. It illustrate how the quotients and random numbers are used for further computation (which we will not discuss here).

```
select Obj_tillaeg_ldel_til_kvvalg (Tmk_land_id, Landsdel_id,
                      Parti_id, Antal_kredsmandater, Kvotient, Random_nr)
  bulk collect into P_col_tillaeg_ldel_til_kvvalg
  from (select   Tmk_land_id,
                 Landsdel_id,
                 Parti_id,
                 Antal_kredsmandater,
                 Tmk_land_kvotient Kvotient,
                 Tmk_random_nr Random_nr
            from (select Tmk_land_id,
                         Landsdel_id,
                         Parti_id,
                         Antal_kredsmandater,
                         Tmk_land_kvotient,
                         Tmk_random_nr,
                         row_number ()
                         over (partition by Landsdel_id, Parti_id
                               order by Tmk_land_kvotient desc)
                           Nr_starttillaeg
                    from (select Tmk_land_id,
                                 Landsdel_id,
                                 Parti_id,
                                 Antal_kredsmandater,
                                 Tmk_land_kvotient,
                                 Tmk_random_nr
                            from (select Tmk_land_id,
                                         Landsdel_id,
                                         Parti_id,
                                         Antal_kredsmandater,
                                         Tmk_land_kvotient,
                                         Tmk_random_nr
                                    from Tillaeg_man_land_kvotienter
                                   where     Valg_id = P_valg_id_in
                                         and Valgfase_kode = P_valgfase_in)))
           where Nr_starttillaeg > Antal_kredsmandater
        order by Tmk_land_kvotient desc,
                 Tmk_random_nr desc);
```

Fig. 4. Procedure Tillaeg_landsdel_hentpotkvot

All tuples from table Tillaeg_man_land_kvotienter are ordered lexicographically, first by the rounded quotient Tmk_land_kvotient (line 23617) and the random number Tmk_random_nr (line 23618), both in descending order.

5 Responsible Disclosure

Denmark has been using a computer program to compute the seat allocations of the Danish Parliament since for at least two decades [6]. The new version of DSAS (studied in this paper) was introduced only after 2007. We identified the rounding problem in DSAS in 2016 and informed the Ministry immediately about our findings. To the best of our knowledge the software as been updated, and the rounding problem has been addressed and fixed.

6 Conclusion

We have shown, that countries that use d'Hondt or Sainte-Laguë methods for computing the final seat allocation of parliament should be aware that rounding quotients in the table may lead to Error 1, *Incorrect Allocation Order*, or even worse, Error 2, *Incorrect Seat Allocation.* We have shown that erroneously rounding can impact an election outcome and that this observation is not just hypothetical, but can with a non-zero probability actually impact real elections. We have also shown that accidental rounding is difficult to detect, after all, on the face of it, how much damage could rounding actually do?

Social choice experts agree that rounding quotients when implementing d'Hondt or Sainte-Laguë methods is a mistake. However, it the election law and relevant election regulations that define the exact rules. The law defines the requirements for seat allocation systems, and if the law requires to round two digits then so be it. The Danish law does is not specific when it comes to rounding, and therefore, it should be the mathematical definition of the voting method that prevails.

Therefore, to implement a d'Hondt or Sainte-Laguë voting rule correctly is easy: One must not store the quotient but instead store both numerator (the Antal_stemmer) and denominator (the V_divisor) in two different fields. Using the following simple rule of arithmetic assuming $b, d \neq 0$

$$\frac{a}{b} < \frac{c}{d} \quad \text{if and only if} \quad a \cdot d < c \cdot b,$$

it is possible to implement the seat allocation for both constituency and compensatory seats without fractions and rounding guaranteeing that the correct seat allocation is computed.

Acknowledgments

The author would like to thank the Danish Ministry for Economic Affairs and the Interior and Statistics Denmark for their support and cooperation. The author would also like to thank Prof. Jørgen Elklit for comments on earlier drafts of this paper.

References

1. Kenneth Benoit. Which electoral formula is the most proportional? A new look with new evidence. *Political Analysis*, 8(4):381–388, 2000.
2. Danmarks Statistik. Folketingsvalget den 13. november 2007. Statistiske Efterretninger, 27. November 2007.
3. Folketing (parliamentary) elections act. Online Report, 2014. Consolidated Act No. 369 of 10 April 2014, translated.
4. Rajeev Goré and Ekaterina Lebedeva. Simulating STV hand-counting by computers considered harmful: A.C.T. In *Electronic Voting - First International Joint Conference, E-Vote-ID 2016, Bregenz, Austria, October 18-21, 2016, Proceedings*, pages 144–163, 2016.
5. Arend Lijphart and Bernard Grofman. Degrees of proportionality of proportional representation formulas. In *Electoral laws and their political consequences*, pages 170–179. Agathon Press, 2003.
6. Rune Pedersen. Unix-veteran kender valgresultatet før alle andre. Computerworld, 13. November 2007.
7. Philip B. Stark and Vanessa Teague. Verifiable european elections: Risk-limiting audits for D'Hondt and its relatives. *USENIX Journal of Election Technology and Systems (JETS)*, 1:18–39, 2014.

Modular Formalisation and Verification of STV Algorithms

Milad K. Ghale, Rajeev Goré, Dirk Pattinson, and Mukesh Tiwari

The Australian National University

Abstract. We introduce a formal, modular framework that captures a large number of different instances of the Single Transferable Vote (STV) counting scheme in a uniform way. The framework requires that each instance defines the precise mechanism of counting and transferring ballots, electing and eliminating candidates. From formal proofs of basic sanity conditions for each mechanism inside the Coq theorem prover, we then synthesise code that implements the given scheme in a provably correct way and produces a universally verifiable certificate of the count. We have applied this to various variations of STV, including several used in Australian parliamentary elections and demonstrated the feasibility of our approach by means of real-world case studies.

1 Introduction

Single Transferable Vote (STV) is a family of vote counting schemes where voters express their preferences for competing candidates by ranking them on a ballot paper. STV is used in many countries including Ireland, Malta, India, Nepal, New Zealand and Australia. It is also used to elect moderators in the StackExchange discussion forum [19] and the board of trustees of the John Muir trust [11].

To count an election according to STV, one usually computes a quota dependent on the number of ballots cast (often the Droop quota [7]) and then proceeds as follows:

1. Count all first preferences on ballot papers;
2. Elect all candidates whose first preferences meet or exceed the quota;
3. Transfer surplus votes, i.e. votes of elected candidates beyond and over the quota are transferred to the next preference;
4. If all transfers are concluded and there are still vacant seats, eliminate the least preferred candidate, and transfer his/her votes to the next preference.

While the scheme appears simple and perspicuous, the above description hides lots of detail, in particular concerning precisely which ballots are to be transferred to the next preference. Indeed, many jurisdictions differ in precisely that detail and stipulate a different subset of ballots be transferred, typically at a fractional weight (the so-called *transfer value*). For example, in the Australian Capital Territory (ACT) lower house STV election scheme, only the *last parcel* of an elected candidate (the ballots attributed to the candidate at the last count) of an elected candidate is transferred. In contrast, the STV variant used in the upper house of the Australian state of Victoria transfers *all* ballots (at a reduced transfer value). Similar differences also exist for the transfer of votes when a candidate is being eliminated.

On the other hand, all variants of STV share a large set of similarities. All use the same mechanism (transfer, count, elect, eliminate) to progress the count and, for example, all cease counting once all vacancies are filled. In this paper, we abstract the commonalities of all different flavours of STV into a set of minimal requirements that we (consequently) call *minimal* STV. It consists of:

- the data (structure) that captures all states of the count
- the requirements that building blocks (transfer, count, ...) must obey.

In particular, we formally understand each single discrete state of counting as a mathematical object which comprises some data. Based on the kind of data that such an object encapsulates, we separate them into three sets: initial states (all ballots uncounted), final states (election winners are declared) and intermediate states. The latter carry seven pieces of information: the list of remaining uncounted ballots which must be dealt with; the current tally of each candidate; the pile of ballots counted in each candidate's favour; the list of elected candidates whose votes await transfer; the list of eliminated candidates whose votes await transfer; the list of elected candidates; and the list of continuing candidates. Basically, they record the current state of the tally computation.

We realise transitions between states, corresponding to acts of counting, eliminating, transferring, electing, and declaring winners, as formal rules that relate a pre-state and a post-state. These rules are what varies between different flavours of STV, so minimal STV does not define them. Instead, it postulates minimal *conditions* that each rule must satisfy. An *instance* of STV is then given by:

1. *definitions* of the rules for counting, electing, eliminating, and transferring;
2. formal *proofs* that the rules satisfy the respective conditions.

We sometimes refer, somewhat informally, to the conditions the various rules must satisfy as *sanity checks*. They are the formal counterparts of the legislation that informs counting officers which action to perform, when. Each sanity check consists of two parts: the *applicability condition* specifies under what conditions the rule can be applied while the *progress condition* specifies the effect of the rule on the state of the count. For example, the count rule is applicable if there are uncounted ballots and reduces the number of uncounted ballots.

We establish three main properties of this generic version of STV. The first is that each application of any of the generic transitions of STV reduces a complexity measure. The second is that at any non-final state of the count, at least one of the generic transitions is applicable because it satisfies its sanity check requirements. The third is that the overall minimal STV algorithm terminates.

All this is carried out inside the Coq theorem prover [3]. Using Coq's extraction mechanism [14] we can then automatically synthesise a (provably correct) program for STV counting from the termination proof. By construction, the executables are certifying programs which produce an visualised trace of computation upon each execution. The correctness of the certificate can be checked by anyone with minimal technical knowledge, independent of the way it was obtained. That is to say, we provably implement the counted-as-cast aspect [4] of universal verifiability. Finally, our experimental tests with real elections demonstrate feasibility of our approach for real world applications. Compared with other formalisations of STV, where even small changes in the details of a single rule requires adapting a global correctness proof, the outstanding features of our work is *modularity*, since sanity checks are local to each rule, and *abstraction*, since the general correctness proof is based on the local conditions for each rule. It is precisely this simplicity that allows us to capture a large number of variations of STV, including several used in Australian parliamentary elections.

2 The Generic STV Machine

We begin by describing the components of minimal STV before discussing their implementation in the Coq theorem prover, together with examples.

2.1 The Machine States and Transitions

The best way to think of minimal STV and its instances is in terms of an abstract machine. The states can be thought of as snapshots of the hand counting procedure, where there is e.g. a current tally, and a set of uncounted ballots at every stage. Tallying is then formalised as transition between these states.

There are three types of machine states: *initial*, *intermediate*, and *final*. An initial state contains the list of all *formal* ballots. Final states are where winners are declared. Each intermediate state consists of seven components:

1. A set of uncounted ballots, which must be counted;
2. A tally function computing the number of votes for each candidate;
3. A pile function computing which ballots are assigned to which candidate;
4. A list of already elected candidates whose votes await transfer;
5. A list of the eliminated candidates whose votes need to be dealt with;
6. A list of elected candidates; and
7. A list of continuing candidates.

We use $\mathcal{C}$ for the set of all candidates participating in an election and use c, c', and c'' for individual candidates from $\mathcal{C}$. We use $\mathsf{List}(\mathcal{C})$ for the set of all lists over $\mathcal{C}$ and use $\mathbb{Q}$ for the set of rational numbers. A ballot is an ordered pair (l, q) where $l \in \mathsf{List}(\mathcal{C})$ is the preference order and $q \in \mathbb{Q}$ is the (possibly fractional) value of this ballot. We write $\mathcal{B} = \mathsf{List}(\mathcal{C}) \times \mathbb{Q}$ for the set of all ballots, i.e. preference ordered lists of candidates, together with a transfer value. We use h and nh for lists of continuing ("$\underline{\text{h}}$opeful") candidates, and e and ne for lists of $\underline{\text{e}}$lected candidates. A $\underline{\text{b}}$ack$\underline{\text{l}}$og is a pair $(l1, l2)$, the lists of elected and eliminated candidates, respectively, both of whose votes await transfer. We use bl, nbl for backlogs, use qu for the $\underline{\text{qu}}$ota for being elected and use st for the number of vacant seats. We use t, nt for tallies and use p, np for piles. The prefix "n" always stands for "new", thus ne is the list of elected candidates in the post state, after an action has been applied.

Suppose that $ba \in \mathsf{List}(\mathcal{B})$, and $bl, h, e, w \in \mathsf{List}(\mathcal{C})$ are given. Assume t is a function from $\mathcal{C}$ into $\mathbb{Q}$, and p is a function from $\mathcal{C}$ into $\mathsf{List}(\mathcal{B})$. We use $\mathsf{initial}(ba)$ for the initial state, use $\mathsf{intermediate}(ba, t, p, bl, e, h)$ for an intermediate state, and use $\mathsf{final}(w)$ for a final one. Having established terminology and necessary representations, we can mathematically define the states of the generic STV machine.

Definition 1 (machine states). *Suppose ba is the initial list of ballots to be counted, and l is the list of all candidates competing in the election. The set $\mathcal{S}$ of states of the generic STV is the union of all possible intermediate and final states that can be constructed from ba and l, together with the initial state* $\mathsf{initial}(ba)$*.*

We now describe the mechanisms to progress an STV count, such as electing all candidates that have reached quota. These steps, formalised as *rules*, are the essence of each particular instance of STV, and are one of the two cornerstones of our generic notion of STV: the other being the properties that rules must satisfy. We stipulate that each instance of STV needs to implement the following mechanisms that we formulate as rules relating a pre-state and a post-state:

start: to determine the *formal* ballots and valid initial states;
count: for counting the uncounted ballots;
elect: to elect one or more candidates who have reached or exceeded the quota;
transfer-elected: for transferring surplus votes of already elected candidates;
transfer-removed: to transfer the votes of the eliminated candidate;
eliminate: to eliminate the weakest candidate from the process; and
elected win: to terminate counting by declaring the already elected candidates as winners;
hopeful win: to terminate counting by declaring the list of elected and continuing candidates as winners.

For the moment, we treat the above as transition labels only, and provide semantical meaning in the next section.

Definition 2 (machine transitions). *The set $\mathcal{T}$ consisting of the labels **count**, **elect**, **transfer-elected**, **transfer-removed**, **eliminate**, **hopeful win**, and **elected win**, is the set of transition labels of the generic STV.*

2.2 The Small-step Semantics

The textual description of STV is usually in terms of clauses that specify what actions are to be undertaken, under what conditions. In our formulation, this corresponds to pre- and post-conditions for the individual counting rules. The pre-condition is an *applicability constraint*: it specifies under what conditions a particular rule is applicable. The post-condition is a *reducibility constraint*: it specifies how applying a rule progresses the count. Taken together, they form the *sanity check* for an individual rule. Technically, applicability constraints ensure that the count never gets stuck i.e. there is always one applicable rule, while the reducibility constraint guarantees termination.

Reducibility. A careful examination of STV protocols shows that each rule reduces the size of at least one of the following four objects: the list of continuing candidates; the number of ballots in the pile of the most recently eliminated candidate; the backlog; and the list of uncounted ballots. Using lexicographic ordering, this allows us to define a a complexity measure on the set of machine states in such a way that each rule application reduces this measure.

Local Rule Applicability. Each instance of STV must, and indeed does, impose restrictions on when rules can, and must, be applied. Most depend on the particular instance, but some are universal. For example, all STV algorithms require three constraints for the elimination rule to apply: there must be vacant seats; there must be no surplus votes awaiting transfer; and no candidate should have reached or exceeded the quota. We constrain each of the counting rules in this way to guarantee that at least one rule can always be applied.

To formulate the sanity checks for each transition, we define a lexicographic ordering on the set $\mathbb{N}^5$ and impose it on non-final states of the generic machine.

Definition 3. *Let* $\{s : \mathcal{S} \mid s \text{ not final}\}$ *be the set of non-final machine states. We define a function* $\mathsf{Measure} : \mathcal{S} \rightarrow \mathbb{N}^5$ *as follows. We let* $\mathsf{Measure\ (initial(}ba\mathsf{))}$ *= (1,0,0,0,0). Suppose* $bl = (l_1, l_2)$*, for some lists* l_1 *and* l_2*, and for a given candidate c,* $\mathsf{flat}\ (p\ c) = l_c$ *where for a list* l *of lists,* $\mathsf{flat}\ l$ *is the concatenation (flattening) of all elements of* l*. Then*

$$\mathsf{Measure\ (state}\ (ba, t, p, bl, e, h)) = (0,\ \mathsf{length}\ h,\ \textstyle\sum_{d \in l_2} \mathsf{length}\ l_d, \mathsf{length}\ l_1, \mathsf{length}\ ba).$$

Note that the first component of the co-domain of the measure function simply reduces measure from the initial state to any intermediate state, and the third component is the sum of the length of the ballots cast in favour of eliminated candidates that await transfer. In the following, we describe the sanity checks for **transfer** and **elect** in detail, and leave it to the reader to reconstruct those for the other rules from the formal Coq development.

Transfer-elected check. The *transfer-elected* rule that decribes the transfer of surplus votes of an elected candidate must satisfy two conditions. The applicability condition asserts that transfer-elected is applicable to any intermediate machine state input of the form $\mathsf{state}([], t, p, bl, e, h)$, where the list of of uncounted ballots is empty, if there are vacancies to fill ($\mathsf{length}(e) < st$), there are surpluses awaiting transfer, ($bl \neq []$), and no continuing candidate has reached or exceeded the quota. Under these conditions, we stipulate the existence of a post-state output which is reachable from input via a transition labelled *transfer-elected*. The reducibility condition requires that any application of *transfer-elected* reduces the length of the backlog bl while elected and continuing candidates remain unchanged. Mathematically, this takes the following form:

Definition 4 (transfer-elected sanity check). *A rule $R \subseteq \mathcal{S} \times \mathcal{S}$ satisfies the* transfer-elected sanity check *if and only if the following hold:*

applicability: for any state input $=$ state$([], t, p, bl, e, h)$ *that satisfies* length*(e)* $<$ *st (there are still seats to fill),* $bl \neq []$ *(there are votes to be transferred) and* $\forall c.\ (c \in h \rightarrow (t\ c < qu))$ *(no continuing candidate has reached the quota), there exists a post-state* output *such that* input *R* output*.*

reducibility: for any machine states input *and* output*, if* input R output *then* input *is of the form* state*([],t,p,bl,e,h),* output *is of the form* state*(nba,t,np,nbl,e,h) and* length*(nbl)* $<$ length*(bl) (i.e. the backlog is reduced).*

The following is immediate from the definition of measure (Definition 3):

Theorem 1. *If transition label R obeys the reducability condition of Definition 4,* input $\in \mathcal{S}$*,* output $\in \mathcal{S}$ *and* input *R* output*, then the* output *complexity* Measure(output) *is lexicographically smaller than the* input *complexity* Measure(input)*.*

Elect check. The action of electing a candidate, also formalised as a rule in our framework, is subject to the following constraints: in the pre-state input $=$ state$([],t,p,bl,e,h)$, where the list of uncounted ballots is empty, some continuing candidate must have reached quota, and there must be a vacant seat. If so, there must exist a post-state output where the set of continuing candidates is smaller, there are still no uncounted ballots, and the piles and backlog for candidates may be updated. Mathematically, this takes the following form:

Definition 5 (elect sanity check). *A rule $R \subseteq \mathcal{S} \times \mathcal{S}$ satisfies the* elect sanity check *if and only if the following two conditions hold:*

applicability: For any state input $=$ state$([], t, p, bl, e, h)$ *and any continuing candidate $c \in h$, if $t(c) \geq qu$ (c has reached quota) and* length$(e) < st$ *(there are vacancies), there exists a post-state* output *such that* input R output.

reducibility: for any states input *and* output*, if* input R output*, then* input *is of the form* state$([], t, p, bl, e, h)$*,* output *is of the form* state$([], nt, np, nbl, ne, nh)$ *and* length$(nh) <$ length(h) *and* length$(ne) >$ length(e)*.*

Analogous to Theorem 1 we have the following:

Theorem 2. *If a transition rule R meets the reducibility condition of Definition 5, then any application of the transition R reduces the complexity measure.*

Similarly we define sanity checks corresponding to other transition labels, namely **start**, **count**, **eliminate**, **hopeful-win**, and **elected-win**. For all such sanity checks, we establish analogues of Theorems 1 and 2. Then by drawing on them, we obtain a corollary on the measure reduction for the generic STV machine.

Corollary 1. *Any transition R corresponding to a machine transition in $\mathcal{T}$ that satisfies the corresponding sanity check reduces the complexity measure.*

2.3 The Generic STV Machine

The sanity checks constrain the computation that may happen on a given input state if the corresponding rule is applied. A set of rules, each of which satisfies the correspoinding sanity check, can therefore be seen as a small-step semantics for STV counting. We capture this mathematically as a generic machine.

Definition 6 (The generic STV machine). *Let $\mathcal{S}$ and $\mathcal{T}$ be the sets of STV states (Definition 1) and transition labels (Definition 2), respectively. The* generic STV machine *is $M = \langle \mathcal{T}, (S_t)_{t\in\mathcal{T}} \rangle$ where S_t is the santity check condition for transition $t \in \mathcal{T}$. An* instance *of M is a tuple $I = \langle \mathcal{T}, (R_t)_{t\in\mathcal{T}} \rangle$, where for each $t \in \mathcal{T}$, $R_t \subseteq \mathcal{S} \times \mathcal{S}$ is a rule that satisfies the sanity check condition S_t.*

In the sequel, we show that each instance of the generic STV machine in fact produces an election result, present a formalisation, and several concrete instances.

2.4 Progress via the Applicability Conditions

One specific "sanity check", in fact the one that inspired the very term, is the ability to always "progress" the count. That is, one rule is applicable at every state, so that the count will always progress, and there are no "dead ends", i.e. states of the count that are not final but to which no rule is applicable. As an example, no rule other than count may apply if there are uncounted ballots (and indeed count must be applicable in this situation), or that all elected candidates shall be declared winners if the number of candidates marked elected equals the number of seats to be filled. The key insight is that if the sanity check conditions (and hence the applicability conditions) are satisfied, we can always progress the count by applying a rule. In a nutshell, the following steps are repeated in order:

- the start rule applies (only) at initial states;
- cease scrutiny if all vacancies are filled by elected candidates ;
- cease scrutiny if all vacancies are filled by elected and continuing candidates;
- uncounted ballots shall be counted;
- candidates that reach or exceed the quota shall be elected;
- the surplus of elected candidates shall be transferred;
- the ballots of eliminated candidates shall be transferred;
- the weakest candidate shall be eliminated.

We realise this order of rule applications in the proof of the rule applicability theorem. We draw upon the local rule applicability property, present in the sanity checks satisfied by the generic STV model, to guide the theorem prover Coq to the proof, according to the pseudo-algorithm above. Hence we formally verify the expectation of STV protocols on the invariant order of transition applications.

Theorem 3 (Rule Applicability). *Let $I = \langle \mathcal{T}, (R_t)_{t\in\mathcal{T}} \rangle$ be an instance of the generic STV machine. For every non-final state* input *, there is a transition label $t \in \mathcal{T}$and a new state* output *such that* input R_t output*.*

Corollary 1 shows that every applicable transition R_t, for $t \in \mathcal{T}$, reduces the complexity measure. Theorem 3 shows that for any non-final machine state, a transition from the set $\mathcal{T}$ is applicable. Jointly, they give a termination property that, in the terminology of programming semantics, asserts that every execution of the generic STV model has a meaning which is the sequence of computations taken to eventually terminate, and that each execution produces an output which is the value of that execution.

Theorem 4 (Termination). *Each execution of every instance of the generic STV machine on any initial state* input *terminates at a final state* output*, and constructs the sequence of computations taken from* input *to reach to* output*.*

3 Formalisation of The Generic Machine in Coq

We have formalised each notion introduced in the previous section in the theorem prover Coq. Our formalisation consists of a base layer, with instances defined in separate modules. The base layer contains the generic inductive types, definitions of sanity checks, parametric transition labels, specification of the STV machine, functions which are used to formulate the generic STV machine, and theorems proved about the generic STV model. It also includes functions which are commonly called by the modules to carry computation for instances of STV. Instances consist of four parts:

1. instantiations of the generic counting conditions defined in the base, with concrete instances of counting rules of a particular STV schemes
2. proofs which establish sanity checks for the instantiated transition rules
3. possibly auxilary fuctions specific to the particular instance of STV
4. an instantiation of the termination theorem which allows us to synthesise a provably correct, and certifiable, vote counting implementation

We now briefly discuss the framework base and explain some design decisions. In the next section, we give modular formalisations of three STV algorithms.

We encode machine states as an inductive type (Figure 1) with three constructors: `initial`, `state`, and `final`. The constructor `state` has six value fields which parametrise the list of uncounted ballots, a list of tallies, a pile function, and lists of backlogs, elected and continuing candidates, respectively.

Tie breaking. To formalise some tie breaking methods used in some STV schemes, we encode tallies into a chronological list so we can trace the number of votes which each candidate received in previous rounds. This allows us to realise one popular tie breaking procedure. In this method, whenever two or more candidates have the least votes, we go backwards stepwise, if need be, to previous states of the machine which we have computed in the same execution, until we reach a state where one candidate has less votes than the tied candidates. Then we update the current state of the counting by eliminating this candidate.

Last parcel. Some STV schemes, such as lower house ACT and Tasmania STV, employ a notion called last parcel, and transfer only ballots included in

```
Inductive STV_States :=
   | initial: list ballot -> STV_States
   | state:   list ballot
           * list (cand -> Q)
           * (cand -> list (list ballot))
           * (list cand) * (list cand)
           * {elected: list cand | length elected <= st}
           * {hopeful: list cand | NoDup hopeful} -> STV_States
   | winners: list cand -> STV_States.
```

Fig. 1. inductive definition of STV machine states

this parcel according to next preferences. Moreover, they compute the fractional transfer value based on the length of the last parcel. In short, the last parcel of a candidate is the set of votes they received which made them reach or exceed the quota to become elected. As a result, we choose to formalise the pile function to assign a list of lists of ballots to every candidate: the element of this list are the ballots that have been counted in favour of the candidate at the successive counts. This allows us to identify precisely the set of ballots that comprise the last parcel of any elected candidate. Consequently, we are able to tailor both the generic transfer and elect rule and instantiations of them in such a way to modularly formalise several STV schemes where last parcel is being used.

Parameters. We formalise the notions of candidates, the quota, and transition labels parametrically. The parameters are later specified in the modules for each particular STV. For example, each transition label is associated with a relation, that is, a function of type `STV_States -> STV_States -> Prop`.

Sanity checks. Corresponding to each generic transition label, there is a formal definition of the sanity checking. Sanity checks are constraints which are expected of every instance of STV to successfully pass in order to be classified as an STV scheme. Here we illustrate the encoding of the sanity checks for the elect transition. Items (1) and (2) in the Figure 2 respectively match with the first and

```
Definition Elect_Sanity_Check (R:STV_States -> STV_States-> Prop)
:=
1. (∀ input t p bl e h, input = state([],t,p,bl,e,h) ->
   ∃ (c: cand),
    length (proj1_sig e) +1 ≤ st
    ∧ In c (proj1_sig h) ∧ (quota ≤ (hd nty t) c) ->
       ∃ output, R input output) ∧
2. (∀ input output, R input output -> ∃ t p np bl nbl e ne h nh,
   input = state([],t,p,bl,e,h)
   ∧ length(proj1_sig e) < length(proj1_sig ne)
   ∧ length(proj1_sig nh) < length(proj1_sig h)
   ∧ output = state([],t,np,nbl,ne,nh))
```

Fig. 2. Sanity check for elect transition

the second items given in Definition 5. Note that the check loosens the constraint so that in order for elect rule to apply, we need an electable continuing candidate and electing them would not exceed the number of vacancies. This allows us to define a concrete elect transition for e.g. CADE STV [2] which elects only one candidate who has reached or exceeded the quota, rather than electing all of the electable candidates together. Moreover, we are able to formalise other instances of elect transitions which do elect all of the eligible candidates in one step.

Generic STV record. We bundle the generic quota, transition labels and the evidence that the generic transitions satisfy the sanity checks in one record type named `STV_record`. For example, one field of `STV_record` is the requirement that the generic elect transition meets the constraints of the elect sanity check, which technically means (`Elect_sanity_check (elect)`) ∈ `STV_record`.

Finally, we formally prove all of the mathematical properties discussed under the previous section for any `stv` of type `STV_record`. In particular, we demonstrate the termination property. The termination theorem is instantiated in separate modules with particular `STV_record` values, such as ACT STV and CADE STV, to obtain termination property for them as well and carry provably correct computations upon program extraction into Haskell.

4 Modular Formalisation of Some STV Systems

We already have discussed some points where STV schemes diverge from one another. They mainly vary in their specification of formal votes, quota, what is the surplus of an elected candidate, how many candidates to elect out of all of those who are electable, how to update the transfer value of votes of an elected candidate, how to transfer the surpluses, or how to eliminate a candidate and then distribute their votes among other continuing candidates. We describe two of the real-world STV schemes we have formalised.

4.1 Victoria STV

The Australian state of Victoria employs a version of STV [20] for electing upper house representatives. Figure 3 depicts the instantiation of the generic **elect** transition label with our formulation of the Victoria STV elect rule. Each line encodes some clauses of the Victorian STV protocol which specify the elect rule. We only explain lines 5, 6, and 7 of Figure 3.

The counting protocol of Victoria STV, defines surpuls votes to be "*the number, if any, of votes in excess of the quota of each elected candidate*". Moreover it dictates, under Section 17, Subsection 7 Clause (a), that "*the number of surplus votes of the elected candidate is to be divided by the number of first preference votes received by the elected candidate and the resulting fraction is the transfer value*". In lines 6 and 7, we compute the surplus vote and the fractional transfer value accordingly and multiply it by the current value of every ballot in the pile of the elected candidate c to update the pile of this candidate.

The protocol further states under subsection (8), and (13) that "*Any continuing candidate who has received a number of votes equal to or greater than the quota on the completion of any transfer under subsection (7), or on the completion of a transfer of votes of an excluded candidate under subsection (12) or (16), is to be declared elected*". The definition requires electing candidate(s) no matter how they have obtained enough votes. We therefore implement clauses (8) and (13) in Line 5, where we elect everyone over or equal to the quota, place them in the update list `ne` of elected candidates, and insist that the list `l` of elected candidates in this state and the old list `e` of elected candidates together form a permutation of `ne`. Insisting that the new (combined) lists of winners are a permutation of `l` and `e` combined also imply that no new candidates are introduced, no existing candidates are deleted, and there is no duplication.

Next, we describe how the updated pile of an elected candidate in `Victoria_Elect` is transferred by Victoria's transfer-elect transition. Figure 4 illustrates the instantiation of the generic transfer-elected rule with a concrete case used by Victoria STV. Notice that in the first conjunct of Line 4 in Figure 3, we order the list of elected candidates according to the tally amount. When it comes to transferring elected surplus, as we see in Line 4 of Figure 4, the biggest surplus is dealt with first which belongs to candidate *c*. Furthermore, Line 5 specifies that *all of this candidate's surplus is distributed* at the fractional value computed in `Victoria_Elect`.

4.2 Australian Capital Territory STV

Lower house elections in the Australian Capital Territory (ACT) use a version of STV [1] which stands out for some of its characteristics, including transfer of the "last parcel" of votes and the formulation of transfer value. The specification of the elect transition of ACT STV is similar to the one in Figure 4 except for lines 6 and 7, which are replaced by the following:

$$\texttt{np(c)=}\ \textsf{map}\,(\textsf{fun}\ \texttt{b =>}\ \frac{(\textsf{fst}\ \texttt{b},\ (\textsf{snd}\ \texttt{b})\times((\textsf{hd}\ \texttt{nty t(c))- quota)}}{(\textsf{Sum}\ \ \textsf{snd}\,(\textsf{last}\,(\texttt{p c}))}\,)\ (\textsf{last}\,(\texttt{p c}))$$

Moreover, the ACT version of transfer-elected is as in Figure 4 except that the fist conjunct in Line 5 is replaced and reads `nba = last (p c)`. The two variations

```
Definition Victoria_Elect input output : Prop :=
∃ t p np bl nbl nh h e ne,
1. input = state([],t,p,bl,e,h) ∧
2. ∃ l, length (proj1_sig e) + length(l) ≤ st
3. ∧ ∀ c, In c l ->(In c (proj1_sig h) ∧(quota ≤ hd nty t (c)))
4. ∧ ordered (hd nty t) l∧ Permutation l(proj1_sig nh)(proj1_sig h)
5. ∧ Permutation l(proj1_sig e)(proj1_sig ne)∧ (nbl= bl ++ l)
6. ∧ ∀ c, In c l -> (np (c) = map(map (fun b ⇒
7. (fst b, (snd b)× ((hd nty t (c))-quota)/((hd [] t)c))(p c)
8. ∧ output = state([],t,np,nbl,ne,nh)
```

Fig. 3. Victoria STV elect transition

```
Definition Victoria_TransferElected input output :=
∃ nba t p np bl nbl h e,
1. input = state([],t,p,bl,e,h) ∧
2. length(proj1_sig e) < st ∧ output = state([],t,np,nbl,ne,nh)
3. ∧ ∀ c, In c (proj1_sig h) -> ((hd nty t) c < quota)
4. ∧ ∃ l c, (bl= (c::l,[]) ∧ (nbl= (l,[])) ∧ (np(c) = [])
5. ∧ (nba= flat(fun x => x)(p c) ∧ (∀ d, d≠c -> (np c)=(p d))
```

Fig. 4. Victoria STV transfer-elected transition

together tell us that we only transfer the last parcel of the elected candidate and the transfer value equals the surplus votes of this candidate divided by the sum of fractional values of this last parcel, rather than the tally of the elected candidate.

There are obvious issues with the transfer value formula used in the ACT STV [10]. For example, it is possible for the calculated fractional value of a surplus vote to exceed 1, which is clearly a flaw of the algorithm. As a result, the software used by the ACT election commission which implements the algorithm [18], makes explicit modifications to ensure no surplus votes exceeds 1. We adapt this corrected version in our formalisation. Nonetheless, nothing would restrict us from selecting the defective original formula of ACT STV, if we chose to.

5 Certifying Extracted Programs and Experiments

We use the built-in mechanisms of Coq to extract executable Haskell programs for each module. The automatic extraction method provides very high assurance that the executable behaves in accordance to its Coq formalisation. Correctness proofs established in the Coq therefore give functional correctness of the executables. However, each execution of the extracted program generates a run-time certificate, providing independently checkable evidence of the underlying computation.

Theorem 4 guarantees each run of the program produces a *formal* certificate, i.e. a sequence of states of the count that are linked by rules, as an element of an inductive data type. (This contrasts with what one may call an *concrete*

```
                 initial [([a,c,b],1/1),([b,c,a],1/1),([c,a],1/1),([c,b,a],1/1)]
----------------------------------------------------------------------------------------------------- start
 state [([a,c,b],1/1),([b,c,a],1/1),([c,a],1/1),([c,b,a],1/1)]; a[0/1] b[0/1] c[0/1]; a[] b[] c[]; ([],[]); []; [a,b,c]
----------------------------------------------------------------------------------------------------- count
state []; a[1/1] b[1/1] c[2/1]; a[[([a,c,b],1/1)]] b[[([b,c,a],1/1)]] c[[([c,a],1/1),([c,b,a],1/1)]]; ([],[]); []; [a,b,c]
----------------------------------------------------------------------------------------------------- eliminate
 state []; a[1/1] b[1/1] C[2/1]; a[[(a,c,b],1/1)]] b[[([b,c,a],1/1)]] c[[([c,a],1/1),([c,b,a],1/1)]]; ([],[a]); []; [b,c]
----------------------------------------------------------------------------------------------------- transfer-removed
 state [([a,c,b],1/1)]; a[1/1] b[1/1] c[2/1]; a[] b[[([b,c,a],1/1)]] c[[([c,a],1/1),([c,b,a],1/1)]]; ([],[a]); []; [b,c]
----------------------------------------------------------------------------------------------------- count
      state []; a[1/1] B[1/1] c[3/1], a[] b[[([b,c,a],1/1)]] c[[(a,c,b],0/1)]]; ([c],[a]); [c]; [b]
      ----------------------------------------------------------------------------------------- elect win
                                       winners [c]
```

Fig. 5. Example of a certificate

certificate which would be a file that comprises a textual representation of the formal certificate.) Moreover, the theorem guarantees that the formal certificate is the sequence of computation performed in the execution to obtain the final result. To produce a concrete certificate from an execution of extracted Haskell program, we need to agree on textual representations for the elements of the data types concerned.

The certificate generated for each input witnesses the correctness of the count. Note that it is trivial to demonstrate that the existence of a correct certificate implies the correctness of the result, as the latter is defined precisely as being obtained through a sequence of correct rule applications. Certificate correctness can be checked by anyone, without *any* trust in the means that were used in the production of the certificate, or the underlying hardware. The fact that concrete certificates can be checked by scrutineers means that our tallying technique satisfies the count-as-recorded property of universal verifiability. Thus any election protocol designed for STV schemes which requires a proof of tallying correctness can use our tool.

Figure 5 illustrates an example of a concrete certificate, where candidates a, b, and c are competing for one seat. We discuss certification only briefly as it is described elsewhere [8]. We use exact fractions for computations to avoid the rounding issues explained in [10]. Every line shows six components, each corresponding to an abstract data representation of the intermediate states of the abstract machine: the list of uncounted ballots; the tallies of candidates; each candidate's pile; the backlog; and the lists of elected and continuing candidates.

We have evaluated the efficiency of our approach by testing the extracted module for the lower house ACT STV on some real elections held in 2008 and 2012 (Figure 6). The Molonglo electorate of ACT is the biggest lower house electorate in Australia, both in the number of vacancies and the number of voters. The extracted program computes the result in just 22 minutes.

electoral	ballots	vacancies	candidates	time (sec)	certificate size (MB)	year
Brindabella	63334	5	19	116	80.6	2008
Ginninderra	60049	5	27	332	128.9	2008
Molonglo	88266	7	40	1395	336.1	2008
Brindabella	63562	5	20	205	94.3	2012
Ginninderra	66076	5	28	289	126.1	2012
Molonglo	91534	7	27	664	208.4	2012

Fig. 6. ACT Legislative Assembly 2008 and 2012

6 A Technical Discussion

We have introduced a framework for formalisation, verification, and provably correct computation with various STV algorithms. In the design decisions that we made, we have been balancing different aspects for designing a framework. The modular design allows for a much simpler realisation than made possibly by other frameworks (e.g. [8]) as we only need to discharge proofs at a per-rule basis which is also reflected in the fact that (a) we capture realistic voting protocols, and (b) we can accommodate a larger number of protocols with ease.

Previous work emphasises data structures and certification, and showcases this by means of monolithic specifications and proofs. Our work adds modularity, and we distil the algorithmic essence of STV into what we call *sanity checks*.

Every instance of STV satisfying the sanity checks enjoys the rule applicability and termination properties established in Theorems 3 and 4. Therefore, for an instance of STV to be verified, we simply need to establish that the sanity checks hold, rather than duplicate the whole proof process. These checks offer an abstraction on the algorithmic side which helps us avoid duplication of code. Unlike previous work, users do not need to know how the application and termination theorems have been proved in order to show termination of their particular instance. Additionally, separation into modules further improves usability. Anyone seeking a verified implementation of their preferred flavour of STV can simply use our framework and instantiate as appropriate.

Figure 7 illustrates the framework architecture. The parameters component includes type level declaration of candidates, vacancies, and the quota. The base comprises the encoding of the generic STV model along with functions commonly called by the dependent modules. The instantiation component consists of instantiating types specified in the parameters file and automatically discharging of required proofs. Finally, the instantiated generic model is called into each module for discharging sanity checks and consequently extracting provably correct Hakell programs.

The ability to just instantiate is significant as many aspects are dealt with once and for all in the base layer of roughly 25000 lines of code. Each module already formalised is less than 500 lines. Therefore, an interested user has to just carry out formalisation and discharging sanity checks in about 500 lines to acquire a verified executable implementation of their favourite STV. On the other

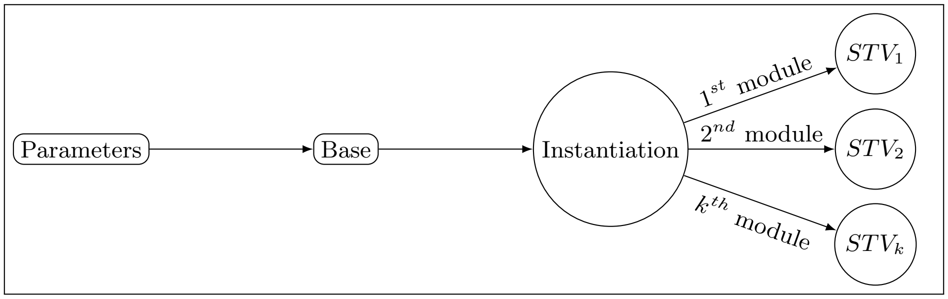


Fig. 7. System Description

hand, accomplishing the same goal by using the previous platform, demands 25000 lines of encoding, along with overcoming numerous technicalities.

Related work. DeYoung and Schürmann [6] formally specify an STV scheme as a linear logic [9] program and then discharge the required correctness proofs inside the logical framework Celf [17]. Celf is capable of executing the specification but their linear logic program does not scale to real-world elections.

Dawson et. al. encode an ML program for STV counting into the HOL4 theorem prover [5] and prove various correctness properties of the program, including termination. HOL4 is able to execute the ML encoding within reasonable time bounds for small elections, but not for large ones. But there is a gap between the HOL4 semantics of ML and those used by ML compilers. This gap could be closed by using the proof-producing synthesis [13] of CakeML code from the HOL assertions, then using the verified CakeML compiler [12] to porduce the machine code. However, this has not been done to date.

Pattinson and Schürmann [15], and Verity and Pattinson [21] formalise a simple version of STV and first-past-the-post elections in Coq and prove properties such as termination and the existence of winners. Then they extract certifying executables in Haskell which can handle real-world elections. Their crucial contribution is that their executable code produces a certificate for every run, which can be idependently verified.

Pattinson and Tiwari [16] extend this method to tackle the Schultz method. Their extracted code handles real-world election and also outputs a certificate for every run. The certificate not only witnesses how the winner was elected, but also provides concrete evidence that each losing candidate is a “loser”.

7 Conclusion

We have designed a modular framework for formalisation, verification, and provably correct computation of STV algorithms. Our work is fully formalised, provides an encoding and provably correct executables for various flavours of STV.

References

1. ACT Legislative Assembly: An act to entrench the principles of the proportional representation (hare-clark) electoral system (1994)
2. Beckert, B., Goré, R., Schürmann, C.: Analysing vote counting algorithms via logic - and its application to the CADE election scheme. In: Bonacina, M.P. (ed.) Proc. CADE 2013. Lecture Notes in Computer Science, vol. 7898, pp. 135–144. Springer (2013)
3. Bertot, Y., Castéran, P., Huet, G., Paulin-Mohring, C.: Interactive theorem proving and program development : Coq'Art : the calculus of inductive constructions. Texts in Theoretical Computer Science, Springer (2004)
4. Cortier, V., Galindo, D., Küsters, R., Müller, J., Truderung, T.: Verifiability notions for e-voting protocols. IACR Cryptology ePrint Archive 2016, 287 (2016)

5. Dawson, J.E., Goré, R., Meumann, T.: Machine-checked reasoning about complex voting schemes using higher-order logic. In: Proc. EVote-ID 2015. pp. 142–158 (2015)
6. DeYoung, H., Schürmann, C.: Linear logical voting protocols. In: Kiayias, A., Lipmaa, H. (eds.) Proc. VoteID 2011. Lecture Notes in Computer Science, vol. 7187, pp. 53–70. Springer (2012)
7. Droop, H.R.: On methods of electing representatives. Journal of the Statistical Society of London 44(2), 141–202 (1881)
8. Ghale, M.K., Goré, R., Pattinson, D.: A formally verified single transferable voting scheme with fractional values. In: Krimmer, R., Volkamer, M., Binder, N.B., Kersting, N., Pereira, O., Schürmann, C. (eds.) Proc. E-Vote-ID 2017. Lecture Notes in Computer Science, vol. 10615, pp. 163–182. Springer (2017)
9. Girard, J.: On the unity of logic. Ann. Pure Appl. Logic 59(3), 201–217 (1993)
10. Goré, R., Lebedeva, E.: Simulating STV hand-counting by computers considered harmful: A.C.T. In: Proc. EVote-ID 2016. pp. 144–163 (2016)
11. John Muir Trust: Apply to be a trustee, https://www.johnmuirtrust.org/assets/000/002/860/How_to_apply_to_be_a_Trustee_Jan_2018_original.pdf, accessed May 15, 2018
12. Kumar, R., Myreen, M.O., Norrish, M., Owens, S.: CakeML: A verified implementation of ML. In: Principles of Programming Languages (POPL). ACM (Jan 2014)
13. Magnus, M.O, Scott, O.: Proof-producing translation of higher-order logic into pure and stateful ML. J. Funct. Program. 24(2-3): 284-315 (2014)
14. Letouzey, P.: A new extraction for Coq. In: Geuvers, H., Wiedijk, F. (eds.) Proc. TYPES 2002. Lecture Notes in Computer Science, vol. 2646, pp. 200–219. Springer (2003)
15. Pattinson, D., Schürmann, C.: Vote counting as mathematical proof. In: AI 2015: Advances in Artificial Intelligence - 28th Australasian Joint Conference. pp. 464–475 (2015)
16. Pattinson, D., Tiwari, M.: Schulze voting as evidence carrying computation. In: Proc. ITP 2017. pp. 410–426 (2017)
17. Schack-Nielsen, A., Schürmann, C.: Celf - A logical framework for deductive and concurrent systems (system description). In: Proc. IJCAR 2008. pp. 320–326 (2008)
18. Software Improvements: Electronic and voting and counting sytems. http://www.softimp.com.au/evacs/index.html, accessed at May 12, 2015
19. StackExchange: Moderator elections (2018), https://math.stackexchange.com/election/6?tab=election, accessed May 15, 2018
20. The Parliament of Victoria: Electoral act 2002 , see also https://www.elections.act.gov.au/education/act_electoral_commission_fact_sheets/fact_sheets_-_general_html/elections_act_factsheet_hare-clark_electoral_system
21. Verity, F., Pattinson, D.: Formally verified invariants of vote counting schemes. In: Proc. ACSW 2017. pp. 31:1–31:10 (2017)

Implementing an audio side channel for paper voting

Kristjan Krips[1], Jan Willemson[1,2], and Sebastian Värv[1]

[1] Cybernetica AS
Ülikooli 2, 51003 Tartu, Estonia
{krisjan.krips,jan.willemson,sebastian.varv}@cyber.ee
[2] Software Technology and Applications Competence Center
Ülikooli 2, 51003 Tartu, Estonia

Abstract. In the ongoing debate between the proponents of electronic and paper voting, a frequently used argument is that electronic voting is susceptible to electronic attacks, and those are less detectable by a human than physical ones. This paper contributes to the research of electronic attacks against paper voting by building a proof-of-concept classifier for audio samples recorded while writing numbers. Such a classifier can be used to break the privacy, for example, in case of preferential voting ballot sheets, or voting systems where the voter must fill in the candidate number. We estimate the quality of the classifier and discuss its implications to the physical security measures of polling stations and ballot design.

1 Introduction

Voting is a form of public opinion polling used when a group of people needs to take a common decision. The size of the group may vary from just a few persons to whole societies, and the decisions may vary from selecting a beauty queen to determining who is going to rule the country for the next 5 years.

The bigger implications the decision has, the more critical role is played by the actual voting and vote counting processes. There are a number of requirements set to contemporary voting systems, and thick rule books describing how to enforce them.

Unfortunately, these rules can be contradictory. In order to gain public acceptance of an election result, all the processes should be fully auditable, ideally by everyone. On the other hand, to prevent coercion and vote-buying, the actual votes should remain secret, introducing an inherently non-auditable component into the system.

It is also the case that important elections tend to have a large voter set easily reaching millions of people. This has implications on the vote counting. A single person is unable to count millions of votes in a reasonable time frame, so this work has to be distributed between many people, not all of whom are equally careful or trustworthy. If a physical medium like paper is used for voting, there can also be ambiguous markings that need interpretation, and this interpretation may depend on the interpreter. And last-but-not-least, organizing voting based on physical carriers is a huge logistical challenge, requiring all of these millions of people to go to polling stations and collecting the ballots later.

These problems have motivated research in alternative vote casting mechanisms, including electronic ones. Starting from T.A.Edison's "Electrographic Vote Recorder and Register"[3], various methods including voting machines and remote vote casting over Internet have been proposed and tried out.

While helping to ease some of the inherent difficulties of elections, electronic means can bring up new concerns. Humans can not control digital environments directly and need to rely on imperfect interfaces. Also, it is hard to be sure that a digital device acts according to its specification and does not include anything extra, like malware.

Another example of out-of-specification behaviour is the existence of side channels threatening vote privacy. Perhaps one of the most notorious examples of potential implications of such problems was observed in the Netherlands. As those events greatly inspired our current research, we will make a short recap here.

1.1 The rise and fall of electronic voting in the Netherlands

Netherlands has been a true pioneer of electronic voting. Legislation allowing machine voting was put in place already in 1965, and the first voting machines appeared in 1966 [8]. The first attempts to automate counting were done in late 1980s. From 1994, the government actively promoted the usage of electronic apparatus in voting [6]. By 2005, the Dutch market had been divided by two bigger suppliers of the voting machines – Nedap and Sdu [8]. There had been a few complaints e.g. favouring a candidate with number 31 due to his/her name being displayed on top of the second column of candidates [6], but in general the public trust in voting machines seems to have been rather high.

However, in 2006, a series of events took place that changed the situation drastically. First, during 2006 elections a fraud suspicion was raised in one of the districts where Nedap voting machines were used. After repeated shadow elections and several rounds in court, this led to a conviction [8].

As a reaction to this (and probably also earlier complaints), a civil activist and hacker Rop Gonggrijp initiated a movement called "Wij vertrouwen stemcomputers niet" ("We don't trust voting computers"). He got access to some of the Nedap machines, managed to reverse engineer the source code and demonstrated the ease of maliciously replacing the onboard chips [6].

The other major problem Gonggrijp and his collaborator Maurice Wessling discovered was the possibility to eavesdrop electromagnetic emanations (called a TEMPEST attack) which, under certain circumstances, revealed the voter's party preferences. More precisely, the name of one of the parties (Christen-Democratisch Appèl) contained a diacritic letter (è) and in order to display this, the voting machine screen had to be switched to a different mode. It was

[3] US patent no. 90,646, patented June 1st, 1869

this switch that could be detected from a distance using rather standard radio equipment [5].

The fix for this problem was straightforward (just use e instead of è), but the authorities also looked at the Sdu machines and the electromagnetic emanation problem was much worse there. In the beginning of 2007, Sdu attempted to recertify its machines, but they managed to deliver a device for testing that did not pass other requirements, so this attempt eventually failed. As a result, in October 2007, the existing regulation allowing voting machines was withdrawn [6]. The Netherlands has been using 19th century paper voting ever since.

1.2 Side channel attacks on voting

As mentioned above, the TEMPEST exploit implemented by Gonggrijp and Wessling falls into the category of side channel attacks. These sorts of attacks are in general relatively difficult to prevent since, by definition, they make use of some out-of-system-model feature like power consumption, message timing, etc.

Electromagnetic emanation leakage is not the first side channel vulnerability considered for voting. Taking a photo of the ballot with a phone or some other device is a well-known privacy problem [2]. Moran and Naor note that in case Direct Recording Electronic (DRE) equipment posts encrypted votes on a bulletin board, posting timing can be used by a compromised DRE machine to reveal the voter preference [9].

An interesting side channel attack (called Three-Pattern) against the ThreeBallot optical scan voting system was described by the original author Ronald Rivest himself [11]. As the voter in this system has exponentially many choices for encoding her vote on the ballot, the coercer may convince her to do so in a predefined pattern, checking later from the public bulletin board that the pattern has been followed. This leakage is actually so severe that, according to Rivest, "...it makes ThreeBallot much less attractive than I had originally hoped for" [11].

Recently, Toreini *et al.* have improved paper fingerprinting techniques. Their approach allows to create short fingerprints of physical paper sheets using off-the shelf apparatus like overhead projector and photo camera with a sufficiently good resolution. As a result, this makes the vote privacy violation attack proposed by Calandrino *et al.* [3] more accessible to a moderately-resourced attacker. This example demonstrates clearly how advancement of technology also makes paper voting more insecure.

In this paper, we will be considering another type of emanation occurring during paper voting, namely the sound that the pen makes while marking the ballot.

The feasibility of extracting (capital) letters from the audio recording was studied by Yu *et al.* in 2016 [13]. Their results are encouraging, but also show significant challenges. If the training data from the attack subjects can be collected in advance and the position of the microphone can be well predicted, the letter recognition precision can achieve almost 65%. However, if the subjects' handwriting can not be studied beforehand, precision drops below 27%. The authors of [13] also extend their attack to recognising words from a predefined dictionary and achieve the best case accuracy of 50-60%.

We will concentrate our efforts on a smaller set of glyphs to recognise, namely Arabic numerals. We will study how well decimal digits can be recognised from the audio samples of writing them, and discuss the implications to voting privacy and ballot design.

The rest of the paper is organised as follows. In Section 2 we will discuss different types of ballot designs and their implications on the vulnerability to audio side channel attacks. Section 3 describes audio sample classification and Section 4 discusses its implications on security of various election settings. Finally, Section 5 draws some conclusions and sets directions for future work.

2 Types of ballots

The primary sources of requirements for the ballot sheet design are local voting traditions and the implied legal requirements. Susceptibility to audio side channels has most likely not been taken into account as a concern. Hence we start our discussion by reviewing some of the typical ballot designs from this viewpoint.

A frequently used ballot type lists a number of candidates and requires marking one or several of them somehow (writing "X" marks next to one's preferences, crossing some candidates out, etc.). Even though audio side channels against such ballot designs are still possible (e.g., the attacker may draw conclusions based on the timings between writing several "X"-s), they require development effort that remains outside of the scope of the current paper.

Good detection accuracy can potentially be obtained for the ballots allowing write-ins, e.g. leaving an empty slot on the ballot sheet to allow voting for an unlisted candidate.[4] As the voters are not forced to write the names in capital letters, recognising each person's handwriting becomes a major problem, and without reliable personalised training data the results can be expected to be considerably worse than those of Yu *et al.* [13].

Still, we can consider a subset of the handwriting recognition problem. For example, in a referendum the participant might be asked to make a binary decision by writing either "Yes" or "No" to the referendum sheet. Such ballots have been previously used e.g. in Australian constitutional referendums and are currently used e.g. in Swiss referendums. We can see that the corresponding ballot design leaks information that can be classified as Yu *et al.* have already shown. Due to the uniqueness of letters and the lengths of the words it should be easy to distinguish between the two cases.

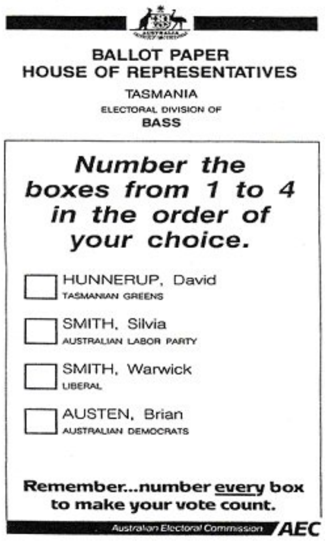
However, there is a specific type of write-ins that has not yet been considered, namely numbers. This is the most promising target of attack for an audio side channel, because the amount of decimal digits is limited to 10, and the variance

[4] This option has been used to cast protest votes. For example, in 1985, Donald Duck received 291 votes in Sweden. As a result, voting for non-existing candidates was prohibited in Sweden starting from 2006: `https://abcnews.go.com/Entertainment/WolfFiles/story?id=91051&page=1`.

of handwritten numbers between different individuals can be expected to be smaller compared to the variance of handwritten letters.

The most common types of ballots where the voter is expected to fill in some numbers come from preferential voting, e.g. single transferable vote (STV) systems (see an example ballot from the Tasmanian House of Representatives elections in Figure 1). Similar kinds of ballots are used, for instance, in:

- Ireland for municipal, parliamentary and European Parliament elections,
- Malta for municipal, parliamentary and European Parliament elections,
- Northern Ireland for European Parliament elections,
- Scotland for municipal elections,
- Austria for European Parliament elections (preference number is optional),
- Australia for electing the Senat and for electing the House of Representatives.



BALLOT PAPER
HOUSE OF REPRESENTATIVES
TASMANIA
ELECTORAL DIVISION OF
BASS

Number the boxes from 1 to 4 in the order of your choice.

□ HUNNERUP, David
TASMANIAN GREENS

□ SMITH, Silvia
AUSTRALIAN LABOR PARTY

□ SMITH, Warwick
LIBERAL

□ AUSTEN, Brian
AUSTRALIAN DEMOCRATS

Remember...number <u>every</u> box to make your vote count.

Australian Electoral Commission AEC

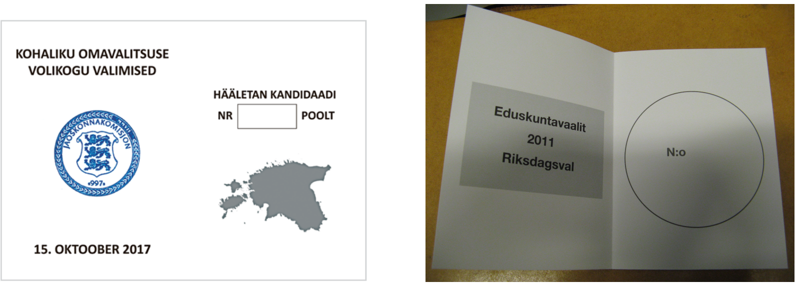
Fig. 1: An example of the Tasmanian election ballot.[5]

When implementing an audio side channel attack against a preferential ballot, we can largely expect to detect two kinds of patterns. First, when we hear the numbers written in the order 1-2-3-4-. . . , the voter is probably filling her preferences in in the ascending order and finding the correct slots on the fly. Without looking at the timings between the numbers, this pattern does not reveal the voter preferences.

[5] Australian electoral systems, `https://www.aph.gov.au/About_Parliament/Parliamentary_Departments/Parliamentary_Library/pubs/rp/RP0708/08rp05`

However, if the voter uses some other order of the numbers, she can be conjectured to fill the ballot from start till the end of the slot sequence, and her preferences leak. This may be expected to be the case with higher probability when the number of slots to fill is smaller.

There are also some countries (e.g. Estonia and Finland) where the voter is expected to write the candidate number on the ballot (see Figure 2). In these cases the audio side channel has the potential of completely breaking the vote privacy.



(a) Ballot used in Estonia for the municipal council elections in 2017 [1]

(b) Ballot used in Finland for the parliamentary elections in 2011. The same ballot design was also used in the 2015 elections.

Fig. 2: Examples of ballots that are designed to be filled with numbers.

The core contribution of this paper is studying the feasibility of identifying the digits by the sound of handwriting. We have created a proof-of-concept implementation that takes an audio sample, splits it into digits and then tries to recognize them. We also created a classifier which performs this task.

The following Section will describe our results in more detail.

3 Audio sample preprocessing and classification

By looking at the waveforms of recordings that correspond to the writing of different digits, it can be observed that the representations of digits are more or less unique. Thus building a good automatic classifier should at least theoretically be possible.

To verify this hypothesis, we conducted several experiments. First, we collected a number of writing samples from volunteers (see Section 3.1 for more details).

Next we tried the standard step of converting the samples into the frequency domain by using fast Fourier transform (FFT). However, if we would only apply FFT, we would get the frequency distribution for the sample, but lose the time dimension. On the other hand, time dimension carries useful information about the digits following the movement of the pen or pencil on the paper. Therefore, we decided to transform the samples into spectrograms. Spectrograms are created by moving a window over the audio sample and applying FFT to the corresponding audio fragments. This gives a representation of the sample where one dimension represents frequency and the other represents time. An example of the result is shown in Figure 3.

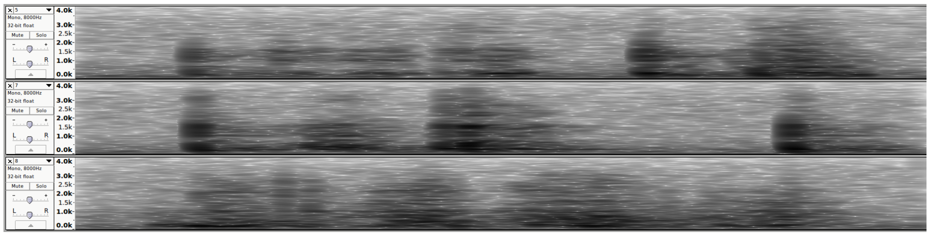


Fig. 3: Spectrogram representations of numbers five, seven and eight.

3.1 Recording and preprocessing

We tested several microphones to find out which one is best suited for the task. The following devices were used: HP laptop, iPhone SE, Jabra Speak 410 and Rode VideoMic Pro. The first three devices had omnidirectional microphones, while Rode VideoMic Pro was a directional cardioid microphone. Comparison of the technical parameters of the microphones is given in Table 1.

Table 1: Comparison of tested recording devices. There was no technical specification available for the microphones in HP laptop and iPhone.

	number of microphones	**type**	**range**	**sensitivity**
HP laptop	2	omni-directional	N/A	N/A
iPhone SE	3	omni-directional	N/A	N/A
Jabra 410 Speak	1	omni-directional	100 Hz - 10 kHz	N/A
Rode VideoMic Pro	1	directional	40 Hz - 20 kHz	-38dB re 1V/Pa ± 2dB @ 1kHz

Testing showed that the laptop microphone was not able to capture handwriting as it could not distinguish the signal from background noise. Rode VideoMic

Pro and the microphone of iPhone SE were able to capture the signal, but the quality was not as good as we got from Jabra Speak 410. It was a bit surprising that the more expensive Rode VideoMic Pro was not able to capture the signal as well as a common conference call device. Therefore, we decided to use Jabra Speak 410 for collecting the training data.

We prepared a sheet of square cells for collecting the samples in order to make the process as uniform as possible. The recording was performed in a closed office room which blocked most of the outside noise. Each volunteer was asked to fill in at least one sheet of ten rows, such that each row would contain all the digits from 0 to 9 once. In addition, the volunteers were asked to leave a small pause after writing each digit to make automatic labelling of the samples easier. The same room and the same table were used for all the samples. The locations of the microphone and the sheet were kept the same throughout the sample collection, with the microphone placed in about 15cm from the edge of the sheet.

Once we had the samples, the next task was to label them to prepare training data for the automatic classifier. As the samples were written on the sheet in a predefined order, we were able to create a script to extract and label the samples. However, manual review of the samples was still necessary to ensure correct operation of the script.

Now that the labelled samples were ready, they had to be prepared for analysis. For that, we converted stereo recording to mono and normalized the tempo. We used WSOLA algorithm [12] to transform the samples such that all of them would have the length of 0.55 seconds. It is important to note that WSOLA does not change the pitch of the sound, otherwise the change of tempo could distort the representation of the digit.

3.2 Building the classifier

We used the k-nearest neighbors algorithm (k-NN) [4] for the classification task. One of the reasons to prefer this method is its capability of producing good results with a small training set. The method works by calculating distance between all samples and then uses majority vote on k nearest samples to determine the class. This was also one of the reasons for normalizing the tempo of the samples as it allowed us to represent the samples as arrays of the same length and therefore align the corresponding frequencies. We pre-processed the data by creating a spectrogram representation from each sample and flattened the output (an array or arrays) to get a one-dimensional array.

We used scikit-learn [10] implementation of the k-NN method to build the model. To use it, the dataset was split into training and testing sets using the `train_test_split` function of scikit-learn. This method allowed us to make sure that the labels would be uniformly distributed in the output sets. The dataset was randomly split into training and test sets so that 10 percent of the samples were used for testing. As the splitting was done on the whole dataset, the ratio of training data to test data did not necessarily hold for the samples belonging to one individual. Thus, individuals might have been over- or under-represented in the training set and test set.

We tested multiple distance metrics to find the one that is most suitable for the representation of the audio data. The results showed that Canberra distance [7] gave significantly better results compared to other distance metrics.

Finally, we used cross-validation for parameter tuning in order to obtain the optimal value of k. We created a list of odd integers as the candidates, fitted a model for each value of k and used cross-validation to determine the k value which gave the best out-of-sample accuracy. In our case, the optimal value for k turned out to be 7.

3.3 Classification results

We used cross-validation to measure the out-of-sample accuracy of the model. Cross-validation partitions the dataset into n equally sized non-overlapping sets, $n-1$ sets are used for training and the n-th set is used for validation. This process is repeated n times, so that each set is validated once. Overall result is calculated by averaging accuracy over all partitions.

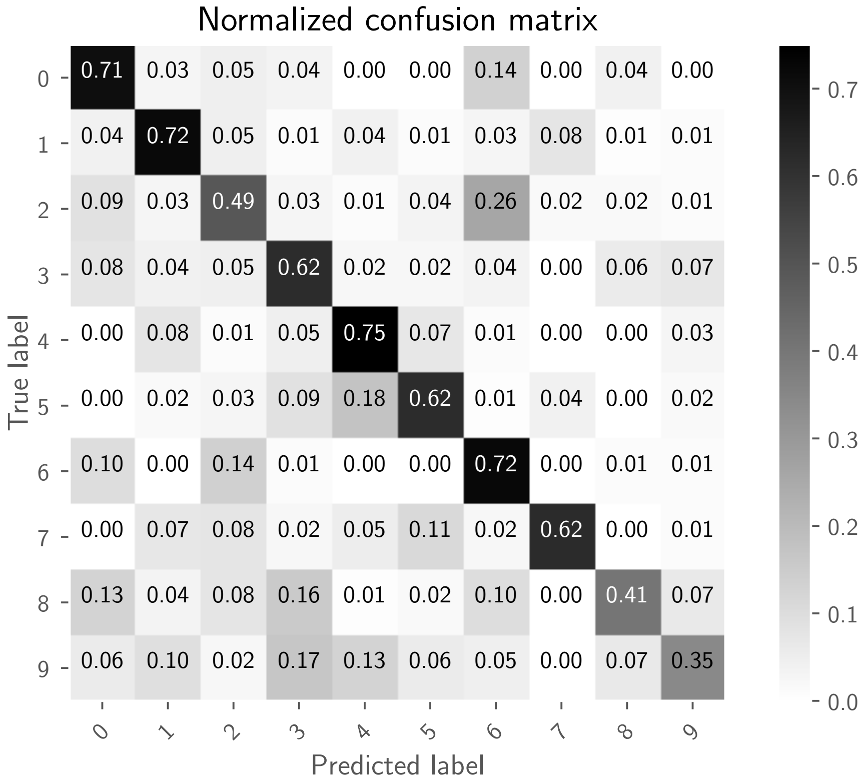
Our dataset consisted of 1676 samples and contained recordings from 11 volunteers. Some of the volunteers contributed more than one data sheet and in one case only part of the data sheet recording was usable due to the corruption of data.

We used scikit-learn implementation of 10-fold cross-validation which uses stratified KFold partitioning strategy. This method provided that uniform number of labels was assigned into each subset. For the classification we used aforementioned k-NN classifier with hyperparameter $k = 7$ as it was previously found to be best suited for our dataset by producing best out-of-sample accuracy. The 10-fold cross validation with the given configuration produced an accuracy of 60.14%. The corresponding confusion matrix can be seen in Figure 4.

We can see from the confusion matrix that the digits 8 and 9 have lower detection accuracy compared to others. One of the reasons for this might be the way how the implementation of scikit-learn breaks ties. Namely, in case of a tie the winner is picked according to the ordering of the classes. Thus, when there is a tie between, say, digits 3 and 8, the first one would win, causing 8 to be determined less.

The low accuracy of 8 and 9 might also be caused by their placement on the data sheet with respect to the microphone. The data sheet was in landscape mode during the recording and the microphone was placed close to the top middle part of the sheet. Therefore, the recorded signal of the digits that were written to the middle of the sheet should have slightly better quality compared to the digits on the sides of the sheet. This reasoning seems not to hold for 0 and 1, but this might be explained by their rather unique audio fingerprint.

Next, we ran a test to find the accuracy for the case when training data is available for the test subject. We took datasets from eleven volunteers and split them into test sets and training sets so that every person contributed 10% of their stratified data points to the test set and the remainder was used for training. Each person had 100 labelled data points and thus 1000 samples were used for training and 100 for testing. Results showed that by using such data on



	0	1	2	3	4	5	6	7	8	9
0	0.71	0.03	0.05	0.04	0.00	0.00	0.14	0.00	0.04	0.00
1	0.04	0.72	0.05	0.01	0.04	0.01	0.03	0.08	0.01	0.01
2	0.09	0.03	0.49	0.03	0.01	0.04	0.26	0.02	0.02	0.01
3	0.08	0.04	0.05	0.62	0.02	0.02	0.04	0.00	0.06	0.07
4	0.00	0.08	0.01	0.05	0.75	0.07	0.01	0.00	0.00	0.03
5	0.00	0.02	0.03	0.09	0.18	0.62	0.01	0.04	0.00	0.02
6	0.10	0.00	0.14	0.01	0.00	0.00	0.72	0.00	0.01	0.01
7	0.00	0.07	0.08	0.02	0.05	0.11	0.02	0.62	0.00	0.01
8	0.13	0.04	0.08	0.16	0.01	0.02	0.10	0.00	0.41	0.07
9	0.06	0.10	0.02	0.17	0.13	0.06	0.05	0.00	0.07	0.35

Fig. 4: A confusion matrix that was created from the output of cross-validation. The accuracy of cross-validation was 60.14%.

average 70.6% of digit predictions were accurate. This result loosely corresponds to the 65% outcome of the experiment by Yu *et al.* [13].

However, the more interesting question concerns usefulness of the classifier when the subject's training data is unavailable. We simulated this situation by selecting one data sheet recording from each of the eleven volunteers. Then we ran eleven tests so that in each test the datasets of ten volunteers was used for training and the data of the volunteer was used for testing the model. Again, training was performed with 1000 samples and the remaining 100 samples were used for validation. The results showed an average accuracy of 49%, with the minimal accuracy of 37% and maximum 65%, respectively. This accuracy can probably improved by collecting more training data.

We also observed an interesting phenomenon during our tests. There was one potential volunteer coming from a completely different cultural background, and the audio samples extracted from his recordings were classified with significantly lower probability.

Visual inspection of his handwriting revealed that this person had a completely different style of writing the numbers, most probably originating from the way numbers are taught in the schools of his country of origin. Thus, in order to achieve good detection accuracy, the volunteers who contribute to the training data should represent the cultural background of the test subjects.

4 Discussion

As expected, detecting digits from audio samples can give better results than that of letters. Compared to about 27% average accuracy of letter detection reported by Yu *et al.* [13], we were able to achieve 49% in the setting where samples from the subject are not available for training.

In the context of elections, the attacker is not typically interested in just one digit, but the whole composition of the ballot. Making use of the fact that several digits need to be written, the attacker may be able to compensate for poor detection of some of them.

For example, in the case of a preferential ballot it is known that all the numbers 1-2-3-... should occur, so if there is one sample that can be interpreted either as 2 or 6 and another one that is definitely 6, we know that the first one must be 2.

Similar reasoning applies for the ballots where the voter needs to write the candidate number. For example in case of Estonia, the candidate number consists of three digits, so the expected correct detection probability is $0.49^3 \approx 0.118$, but not all of the possible triplets correspond to existing candidate numbers. Note that the audio side channel can also be used to detect which candidates the voter *did not* select with high probability. This information may be of equal interest for the attacker in the coercion setting.

Success of the audio side channel attack in the setting of paper voting directly depends on the quality of the audio samples the attacker is able to capture. This quality in turn depends on several aspects: amount of background noise, quality of the microphone and the ability to place the microphone into a good location.

Adding more noise in the polling station does not work as a good countermeasure, since it may have a general irritating effect on the voters. In case the level of the background noise is low, our experiments show that already a mid-class microphone can get relatively good results.

Hence, the main success factor that both the attacker and defender can influence is the microphone placement.

We have conducted no research on the physical protection measures of polling stations, but we conjecture that these measures mostly do not take the threat of audio surveillance into account. There are several strategies the attacker may use to plant the microphones into the voting booths. He may try to access the booth tables in the storage before elections, or assume the role of a voter himself, entering the booth to both mark his own ballot and to leave a microphone there.

Assuming physical access to the voting booths, a similar attack of planting video recording equipment is conceivable. Contemporary cameras also have miniature size; however, they require a direct line of sight, restricting the choice of potential locations. We have not studied the effect of microphone placement extensively, but our testing shows that the signal one gets when attaching a microphone under a wooden table is actually pretty strong and clear.

The only reasonable countermeasure against audio side channel attacks is regular inspection of the voting booths during the elections to detect illegitimate recording equipment. In principle, changing the ballot designs to avoid write-ins could also help, but this may require changing the whole voting tradition and may hence not work in practice. Also, alternative designs (like marking some candidates with "X"-s) may be vulnerable to other side channel attacks of timing, triangulating the locations of the marks, etc. Studying such side channels is an interesting avenue for future research.

And last-but-not-least we would like to emphasize that the privacy-leaking side channel is inherently an issue of paper-based elections, and, to an extent, less so in case of remote electronic voting. Of course, one can imagine video recording equipment installed in someone's home, but such an attack would scale much worse than planting a microphone in a polling booth.

Thus, the main wide-scale privacy attack vector against Internet voting would still require using specially crafted malware.

Note that just an attack against vote privacy is not very interesting on its own, it becomes a real problem in conjunction with coercion. Coercion, in turn, implies the need to target specific voters.

The ease of installing malware on the computers of a particular set of target persons may depend on many aspects like physical security of their homes and general level of digital hygiene. However, we argue that determining the polling station where the target group goes voting and planting microphones there is an attack of lower technical complexity.

Planting the recording equipment can be performed by a corrupt voter (who may be the attacker himself or a voter bribed by the attacker). The attacker may then remain in the polling station observing the times when the voters enter the booth. The recording equipment, in turn, may save time stamps of the collected writing samples, and the time stamps can later be cross-referenced with the times recorded by the observing attacker. Alternatively, the recording equipment may have radio communication capability, reporting the recordings as soon as they have been detected.

Note that this attack requires significant human involvement as the attacker would need to visually identify the voters who enter the booth. However, this step can also be automated by using facial recognition software together with a corresponding personalized facial features database. At the time of this writing (summer 2018), such databases are probably not yet available for medium-level attackers, but they are being built by intelligence organizations based on vast amount of personal images available via social networks.[6] It is only a matter of time when such databases can be bought on black markets.

[6] https://www.forbes.com/sites/thomasbrewster/2018/04/16/huge-facebook-facial-recognition-database-built-by-ex-israeli-spies/

We stress again that our final argument is made only about vote privacy violations via side channel leakages, and does not seek to compare security of paper and remote electronic voting otherwise.

5 Conclusions and further work

There are entire communities devoting their efforts to proving superiority of paper voting over its electronic counterpart (like `https://www.verifiedvoting.org/` and `http://handcountedpaperballots.org/`). An important argument used in such efforts is that high-tech solutions are vulnerable to high-tech attacks, and the latter ones are not yet understood well enough to provide satisfactory mitigation measures.

What proponents of such arguments often do not mention is that high-tech methods can also be used against low-tech elections. The current paper stressed this point by presenting an audio side channel attack against the form of paper voting where the voter is expected to fill in the ballot by writing some numbers.

Success of such an attack in practice depends on many aspects like noisiness of the polling station and the ability to place microphones well enough to capture good-quality audio samples. However, we argue that the resulting leakage is considerably more severe than that of the TEMPEST attack by Gonggrijp and Wessling that forced all electronic voting initiatives in the Netherlands to halt in 2007. Our attack has the potential of revealing the exact voter preference, whereas the attack by Gonggrijp and Wessling only leaked whether the vote was given to one specific party (CDA) or not.

We are not claiming that all the paper voting should be discontinued, but we do advocate for balancing the criticism against electronic voting based on the problems that actually exist in the case of paper voting as well. Our research also implies that side channel attacks should be taken into account while designing the ballot sheets and planning physical protection measures in the polling stations.

This paper presented an attack on a rather specific form of paper voting. However, there are also many other designs of ballot sheets that deserve attention from the viewpoint of advanced technological attacks as well. This remains the subject for future research.

Acknowledgements

The research leading to these results has received funding from the Estonian Research Council under Institutional Research Grant IUT27-1 and the European Regional Development Fund through the Estonian Centre of Excellence in ICT Research (EXCITE) and the grant number EU48684. We would also like to thank all the volunteers contributing the writing samples used in this research, anonymous reviewers for their comments, and our shepherd Dr. Marco Prandini for helpful and thought-provoking discussions.

References

1. Kohaliku omavalitsuse volikogu valimiste käsiraamat 2017. `https://www.valimised.ee/sites/default/files/uploads/kov2017/KOV2017_kasiraamat_web.pdf`.
2. Ben Adida and C Andrew Neff. Ballot Casting Assurance. In *USENIX Electronic Voting Technology Workshop*, 2006.
3. Joseph A Calandrino, William Clarkson, and Edward W Felten. Some Consequences of Paper Fingerprinting for Elections. In *EVT/WOTE*, 2009.
4. Thomas M. Cover and Philip J. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, September 1976.
5. Rop Gonggrijp and Willem-Jan Hengeveld. Studying the Nedap/Groenendaal ES3B Voting Computer: A Computer Security Perspective. In *Proceedings of the USENIX Workshop on Accurate Electronic Voting Technology*, Berkeley, CA, USA, 2007. USENIX Association.
6. Bart Jacobs and Wolter Pieters. Electronic Voting in the Netherlands: from early Adoption to early Abolishment. In *Foundations of security analysis and design V*, pages 121–144. Springer, 2009.
7. G. N. Lance and W. T. Williams. Computer programs for hierarchical polythetic classification ("similarity analyses"). *The Computer Journal*, 9(1):60–64, 1966.
8. Leontine Loeber. E-voting in the Netherlands; from general acceptance to general doubt in two years. In *3rd international Conference on Electronic Voting*, volume 131 of *GI-Edition Lecture Notes in Informatics*, pages 21–30. 2008.
9. Tal Moran and Moni Naor. Receipt-Free Universally-Verifiable Voting with Everlasting Privacy. In Cynthia Dwork, editor, *Advances in Cryptology – CRYPTO 2006*, volume 4117 of *LNCS*, pages 373–392, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
10. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
11. Ronald L Rivest. The ThreeBallot Voting System, 2006. `http://people.csail.mit.edu/rivest/Rivest-TheThreeBallotVotingSystem.pdf`.
12. W. Verhelst and M. Roelands. An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech. In *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 554–557 vol.2, April 1993.
13. Tuo Yu, Haiming Jin, and Klara Nahrstedt. Writinghacker: Audio based eavesdropping of handwriting via mobile devices. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 463–473. ACM, 2016.

On Trade-offs of Applying Block Chains for Electronic Voting Bulletin Boards

Sven Heiberg[1], Ivo Kubjas[1], Janno Siim[3,4], and Jan Willemson[2,3]

[1] Smartmatic-Cybernetica Centre of Excellence for Internet Voting
Ülikooli 2, 51003 Tartu, Estonia
{sven,ivo}@ivotingcentre.ee
[2] Cybernetica AS, Ülikooli 2, 51003 Tartu, Estonia
janwil@cyber.ee
[3] Software Technology and Applications Competence Center
Ülikooli 2, 51003 Tartu, Estonia
[4] Institute of Computer Science, University of Tartu
Ülikooli 18, 50090 Tartu, Estonia
jannosiim@gmail.com

Abstract. This paper takes a critical look at the recent trend of building electronic voting systems on top of block chain technology. Even though being very appealing from the election integrity perspective, block chains have numerous technical, economical and even political drawbacks that need to be taken into account. Selecting a good trade-off between desirable properties and restrictions imposed by different block chain implementations is a highly non-trivial task. This paper aims at bringing some clarity into performing this task. We will mostly be concentrating on public permissionless block chains and their applications as bulletin board implementations as these are the favourite choices in majority of the recent block chain based voting protocol proposals.

1 Introduction

In virtually all of the modern democratic societies, democracy (translated from Greek roughly as *rule of the people*) is implemented via some sort of public opinion polling e.g. voting on the elections of representative bodies.

Regrettably, the process of public polling is very fragile. Many things can go wrong, and have gone wrong in the history of elections. To address the historically experienced problems, many requirements have been put forward and many measures have been developed to meet them. Contemporary democratic nations have thick rule books describing how to run elections so that the number of problems would not rise above the threshold where general public would start questioning legitimacy of the elected bodies.

On the other hand, extensiveness of the rule book is actually a problem of its own. Having many (possibly even contradictory) requirements makes it difficult to make sure all of them are followed, which in turn translates to decreased transparency of the whole process.

The main measure to improve the transparency of election processes is to make them more publicly auditable. In case of electronic voting, this can only be achieved if as much data about the voting as possible can be accessed by public observers. In the end of the auditing procedures, the observers should agree on the outcome, which presumes that they also must be given the same input to start with.

Presenting a uniform view on some digital assets to several independent parties is a surprisingly hard task known as the problem of setting up a *bulletin board*. The first theoretically sound proposals to solve this problem for electronic voting have emerged only in the recent years [11,10,19], and they are relatively complex to implement. For example, an election organiser using protocol by Culnane and Schneider [11] needs to find four independent participants to achieve adversarial tolerance against one dishonest party. With public permissionless financially incentivised block chains the distributed ledger infrastructure already exists and the required organisational and technical effort to use it is intuitively less than setting up a new election specific bulletin board.

Block chain technology has been identified as a useful tool in order to address various auditing challenges. Already in 2007, Sandler and Wallach described the idea of hash linking of votes to guarantee their integrity [35], applying it later in the VoteBox system [34]. In 2011, Benaloh and Lazarus proposed a similar approach to mitigate their Trash attack [6], and in 2013 Bell *et al.* used the same idea as a part of STAR-Vote system [5].

In recent years, the number of similar proposals has risen considerably, and block chains have been pushed as an almost miracle solution to integrity problems. There are numerous academic [41,23,26,29,7,42,20,8,33,12,21,39] and market-oriented[5] proposals aiming at bringing block chains into voting processes. Unfortunately, the level of information provided about these initiatives (especially market-oriented ones) varies a lot and is often limited.

The current paper aims at putting together a higher-level view on different aspects of using block chain technology for electronic voting. Note, however, that we are not targeting a fully systematic treatment of the topic and keep the approach somewhat informal.

The paper is organized as follows. Section 2 makes a short introduction to block chains and their proposed usages for electronic voting. Section 3 points out the main shortcomings not very well addressed by the current proposals. Next, Section 4 points out the main tradeoffs to decide upon while building a block chain based voting system. Finally, we give our conclusions in Section 5.

[5] Some examples active at the time of this writing, summer 2018, include Follow My Vote `https://followmyvote.com/`, Polys `https://polys.me/`, SecureVote `https://secure.vote/`, VoteWatcher `http://votewatcher.com/`, Agora `https://agora.vote`, e-Vox `http://e-vox.org/`, TIVI `https://tivi.io/`, Boulé `https://www.boule.one/`, Democracy.Earth `https://www.democracy.earth/`, Voatz `https://voatz.com/`, Coalichain `https://www.coalichain.io/`, etc.

2 Block chains

The concept of a block chain does not have a single, universally agreed upon mathematical definition. However, different implementations seem to have a few common points.

- Data storage occurs in *blocks*, where the exact content of a block or its semantics may vary (e.g. it may contain transactions for cryptocurrency applications).
- The blocks are linked into a sequence (also called a *ledger*) using a cryptographic hash function.

The idea of hash linking data items is not at all new, going back to at least early 1990s to the works of Haber, Stornetta *et al.* on digital time stamping [16,4]. However, it seems to be exactly this idea of hash linking that gives block chains the attractive property of integrity assurance, since cryptographic hash functions are supposedly hard to invert, making it difficult to revert the linking once it has been performed.

The real renaissance of block chains happened in late 2008, when a researcher (or a group of researchers) hiding behind the pseudonym Satoshi Nakamoto published what is nowadays known as Bitcoin white paper [28]. Essentially, Nakamoto showed how to use available cryptographic and networking tools to achieve a new type of decentralized consensus protocol.[6]

The core innovation of Nakamoto's proposal is introducing computationally difficult puzzle solving (proof of work) together with financial incentives to consensus building. Whoever solves the puzzle first can create the next ledger block and is rewarded with a certain amount of bitcoins. Due to some similarity with gold mining, the participants in this joint effort are called *miners* or *mining nodes.*

Nakamoto's original motivation was to build a monetary system and there the need for consensus is clear – value exchange can only function correctly when there is a universally accepted way of deciding who has how much money.

However, the problem of obtaining a coherent view on the system in a distributed manner is more general, and this is why the original Bitcoin protocol and infrastructure have been used for a myriad of alternative applications, including voting.

It is worth noting that the original Bitcoin white paper does not present any formal definitions of targeted properties, and contains only a simplified security analysis. Follow-up work by Garay *et al.* [14] and Pass *et al.* [31] have formalized several aspects of block chains and clarified the necessary assumptions to prove the security of Bitcoin protocol.

[6] The origin of the term "block chain" is somewhat unclear. It seems to haver been used in some cryptography-related mailing lists in mid 1990-s, but the first occurrence is hard to track. It is interesting to note that Nakamoto's white paper only uses the term "chain of blocks" and not "block chain".

Another functionality making block chains appealing for legal applications is the ability to run smart contracts. Originally proposed already in mid-1990-s by Nick Szabo [37,38], smart contracts can be though of as a scripting layer on top of a block chain, allowing to check fulfillment of certain conditions, and enforcing predefined actions in the respective cases. There are several block chain frameworks that offer this functionality in a form of a programmable execution environment, including Ethereum Solidity[7], Hyperledger Fabric[8] and Cardano Plutus[9].

In principle, it is possible to formulate any set of rules (say, defining correctness of voting or tallying) in the language of smart contracts. In practice, however, the performance requirements needed to actually run them may become prohibitive. We will come back to this issue in Section 3.5.

Block chains come in several flavours. Bitcoin block chain is an extreme example of a distributed ledger where there is no single trusted entity to coordinate the work, nor to decide which blocks to accept from whom, etc. In this case we speak of a *permissionless ledger*.

However, this is not the only option. It is also possible to set up a block chain where data commitments are only accepted from a predetermined set of nodes, and there may even be an authority deciding that some of the blocks will not be admitted. Such a ledger is called *permissioned*. Block chains built within the Hyperledger framework are examples of such a paradigm.

Similarly, it is not necessarily the case that anyone is given access to the block chain for reading. Depending on whether or not general access is allowed, we speak of *public* or *private* block chains, respectively.

For the most part of this paper we will be treating public permissionless ledgers, and there are several reasons for that. First, such ledgers aim at building a fully distributed consensus mechanism which seems to be an attractive property for electronic voting systems. Second (and probably implied by the first reason), majority of the proposals we have studied in course of this research build on top of public permissionless ledgers (mostly Bitcoin or Ethereum). However, several of our observations hold for other kinds of block chains as well.

2.1 How to use block chain for electronic voting?

The obvious application for a block chain in electronic voting is to use it as a bulletin board for committing the state of an electronic ballot box. Dedicated bulletin board protocols for electronic voting do exist [11,10,19], but the assumptions made for achieving the security target are expensive to fulfil. For example, the protocol presented by Culnane and Schneider [11] requires correct behaviour by strictly more than 2/3 of the peers. Hence, in order to tolerate one malicious party, at least three honest peers are required. An election organiser willing to apply such a bulletin board protocol for integrity and transparency

[7] `https://ethereum.org/`

[8] `https://www.hyperledger.org/projects/fabric/`

[9] `https://cardanodocs.com/technical/plutus/introduction/`

reasons would need to find a significant number of independent participants to provide adversarial tolerance. Note that the situation is different when setting up a fault tolerant distributed storage where all nodes actually could be hosted by a single organization and same personnel. With bulletin board we must take into account that some nodes may be malicious. If a single entity is running the bulletin board, then we get no adversarial tolerance against this entity.

In case of a public permissionless financially incentivised block chain there is no problem of finding independent participants – the distributed ledger infrastructure exists and its security is maintained by a number of parties. It is only a matter of finding good use of this infrastructure for election purposes. There is a remarkable number of proposals suggesting variety of approaches involving also (but not only) block chain as a bulletin board.

- It is possible to utilise smart contracts to enforce voting rules [25,33,21,2].
- Several proposals implement vote casting via Bitcoin cryptocurrency transfer [7,42,41,29,26,23,8]. However, they differ a lot in implementation details, e.g. eligibility verification and voter authentication (see Section 3.1).
- To enhance voter anonymity, Takabatake *et al.* propose using Zerocoin instead of Bitcoin [39].
- Block chain based bulletin board can be used to directly commit votes, like e.g. in the schemes of Polys [3] and Agora [13].
- A related approach is taken by the TIVI framework where block chain is used for digital time stamping of certain integrity-critical events, e.g. vote submission [1].
- A few US-based initiatives like VoteWatcher and Votebook are still paper voting systems at their core, but use either a public or private block chain for committing certain data required for later verification (e.g. scanned paper ballots) [30].
- A weaker version of the last approach was used by the Agora team in Sierra Leone where they typed in the votes read out loud by the election officials, and used block chain to verify the count.[10]

These approaches can also be combined. For example one may commit all the votes to a private block chain and make commitments to a public ledger like Bitcoin from time to time; such a solution is implemented e.g. in Agora and VoteWatcher.

There are also other voting systems that could potentially make use of controlled bulletin boards. For example, there is a Vote Registration Service compo-

[10] `https://medium.com/agorablockchain/agora-official-statement-regarding-sierra-leone-election-7730d2d9de4e`. The 2018 Sierra Leone event was advertised by the Agora team as the “world’s first ever blockchain elections” in their press release `https://agora.vote/pdf/Agora_Press-release_SL2018.pdf`. However, the mode that the block chain was eventually used in offered little to no advantages over a simple independent Excel-aided recount. After this disclosure, the Agora team took down their press release, but one can still find a copy of it cached by Google.

nent in the Estonian IVXV scheme [18] which can in principle be implemented on top of a private or public block chain.

In the next Section, we will discuss some of the common concerns not very well addressed in various proposals.

3 Shortcomings of block chain based voting systems

We argue in the following that there are several problems and limitations with using block chain in electronic voting.

3.1 Eligibility verification

Even though the overall target of block chain based voting systems is increasing transparency and public verifiability, there are several aspects of elections, correctness of which can not be established on the block chain. One of the prominent examples is deciding the eligibility of voters.

This problem manifests itself clearly in case of the proposals where Bitcoin transactions are used for vote casting [7,42,41,29,26,23,8,39]. The original design goal of Bitcoin as a cryptocurrency was to provide anonymous transfers, and this is something that contradicts the needs of voting. Even though we typically want the votes to be secret, the voters should still be uniquely identifiable in order to determine eligibility and provide uniformity (so that no-one would get more than one vote).

Hence an identity provider is required one way or another. Note that whoever that provider is, it has the ability to flag ineligible persons as eligible or vice versa, or even define new virtual voters who do not have a corresponding physical person [39]. The only setup where this problem can be ignored are small-scale boardroom type of elections where all the voters know each other. But for even a moderate size elections this can lead to a ballot box stuffing attack that is undetectable by any verification mechanism that may run on top of the block chain. Hence block chain does not remove the need for external trust anchors.

An interesting approach to identity validation in a distributed manners has been taken by the Democracy Earth Foundation. Their manifesto [2] proposes participant registration via creating a video where enough personal details are stated, and seeking acceptance to this video from the community. While this approach may follow the spirit of decentralized governance, it is hard to imagine such an identity creation mechanism to be accepted for official national elections any time soon. Crowd-sourcing-based identity providers are only applicable in small-scale community settings.

Even if we accept the need for external trust dependency for eligibility verification, the problem of implementing the link from identity provider to the block chain still remains. Different frameworks have different approaches to tackle this issue.

Wu proposes using ring signatures to provide anonymity [41], but this introduces a non-trivial setup procedure and a significant performance penalty, making the solution unusable even for moderately-sized elections.

Noizat requires each candidate to issue a key pair to every voter, organizing the public parts into a Merkle tree. Two more keys per voter are generated by the other components of the system, and to cast a vote, a 2-of-3 multisignature scheme is used [29]. However, this non-trivial cryptographic machinery still does not resolve the problem of eligibility verification, but actually makes it worse, forcing all the candidates to manage voter lists.

Lee *et al.* acknowledge the need to have Trusted Third Party (TTP) for identity confirmation, but to implement this procedure they only propose a password-based registration and authentication mechanism [23]. As a result of authentication, the user gets a confirmation that her asymmetric key pair is declared eligible, but the overall system is only as secure as the original, user-created password. Also, there is a strong reliance on the honest behavior of TTP. Dishonest TTP could easily manipulate the result, for example, by claiming that votes for a candidate whom the TTP does not like came from unregistered accounts.

Bistarelli *et al.* propose using Anonymous Kerberos protocol for voter authentication [7]. This approach has the benefit of relying on a relatively standard authentication mechanism that does not impose overly restrictive performance limitations. However, voter anonymity can be broken when Authentication Server and Token Distribution Server collude. More servers can be added to address this issue, but this would make the protocol more complex. Still, out of all the proposals we have studied, the one by Bistarelli *et al.* seems the most viable.

Zhao *et al.* [42] have developed a group incentive mechanism to motivate a group of voters to participate in tallying, but they completely ignore the problem of eligibility verification. Similarly, the voter identification problem is ignored by the considered frameworks that use block chain as a generic bulletin board implementation, including Polys [3] and Agora [13].

There is also a more general problem common to several of the approaches described above.

Namely, elections may last over a longer period of time (say, a week). Eligibility status of potential voters may change during this period (e.g. someone may turn 18 or die). It is, of course, possible to ignore this problem and only let people eligible at a certain point of time to vote [39]. However, another conceivable viewpoint is that it would be more fair to update the list of voters as the changes in eligibility occur.

For small-scale elections with a public voter list this issue can, in principle, be solved by committing the full voter list together with the potential updates to the ledger. But for larger events with potentially non-public sets of voters, only the schemes of Lee *et al.* [23] and Bistarelli *et al.* [7] have some potential to address this issue as they are using an online protocol with the identity provider as a backend service.

3.2 Ensuring ballot secrecy

In addition to integrity, voting protocols also have confidentiality requirements. Voting in majority of the elections is carried out by secret ballot, whereas the verifiability of the correctness of the final tally is still a desired property. Additionally – the tally must not be available before a fixed moment in time. Election result is public information, once it has been released, at the time of the voting the partial results are considered confidential, because of the potential impact on the voter behaviour.

Public block chain contents are public by definition, anybody who hosts a full Bitcoin or Ethereum node has access to all the data published there. The first implication of the fact is that unless the ballots on the block chain are obfuscated, anybody has access to the partial results, thus violating the common requirement of not releasing partial results too early.

Obfuscated ballots (via public-key encryption or some commitment scheme) on the block chain introduce the question of verifiable de-obfuscation. The block chain is of no use if we cannot verify the consistency of the tally. Note that in case of encryption, we cannot store the private key on the block chain, and some external tallying authority will be needed at least for the key management if not for the verifiable tally.

Self-tallying voting protocol Open Vote Network [17] (OVN) has been implemented as an Ethereum smart contract [25]. This protocol does not rely on any third parties for the tally, ballot secrecy is ensured jointly by all voters who have to participate in both obfuscation and de-obfuscation in order for the tally to be successful. The protocol provides privacy for the voters and benefits from the block chain as a public broadcast channel making it an excellent protocol for smart contracts. The downside of the OVN is its fragility – even one voter can prevent tallying simply by being absent. The properties of the protocol make it usable for small scale elections in a boardroom environment, but not for any large scale elections.

3.3 Consistency verification

The common characteristic of all the proposed block chain voting protocols is committing votes (in a plain or encrypted form) to the ledger. Committing just for the sake of it does not make any sense, hence there should exist a routine of checking certain claims about the commitments.

What these claims exactly are depends largely on the scheme. They can include zero-knowledge proofs of correct vote formatting, correspondence to some eligibility criteria, signature validity, block chain integrity, etc. Hopefully there is a relatively short list of them so that the corresponding checks can be implemented with a reasonable amount of code. On the other hand, none of the block chain voting scheme proposals we considered has claimed a full list of checks needed to establish internal consistency of the ledger. The most detailed attempt to describe consistency rules was made for Votebook by Kirby *et al.*, but even they only state that there should be sufficient information released to the public so that "blockchain can be counted correctly" [20].

There are several aspects to consider when defining internal consistency rules of a ledger and determining the final tally outcome.

One problem is dealing with simple, honest (or dishonest) mistakes. For example, due to a programming error, network delay or any other stochastic issue some of the data items required for later auditing may be missing or malformed. The originally desired property of ledger immutability suddenly becomes an issue, since one can not simply replace malformed items or add missing ones to where they should have been. Hence there must be an option of adding data blocks with exception handling and overriding semantics to the ledger, and the auditing logic must be capable of dealing with them. Taking all kinds of potential options of malformedness into account, consistency verification may become complicated beyond what one feels comfortable with.

The problem of unforeseen situations is clearly acknowledged by the Agora team [13]. As a resolution, they suggest using predetermined human auditors who have a power of any ruling they see fit to settle the matter. The ruling should be written down, signed and committed to the ledger. As the unforeseen situations do not follow any patterns, the signed statement is probably also human text that is then in turn open to misinterpretations, defying the purpose of block chain transparency. An interesting open question is whether such statements could be issued in the form of smart contracts.[11]

Another problem is managing repeated vote submissions. Depending on the setup, this may be either a necessary anti-coercion measure (see e.g. [18]), or an unwanted side effect of remote voting. Either way, some rules are necessary to determine which one of the submitted votes will be counted [41,23].

Smart contracts could be used to efficiently describe audit logic and prevent stochastic errors as the ones described above, but they come with a price. Namely, one has to spend a certain amount of resource (e.g. *gas* in case of Ethereum [40]) for the transactions, where the exact amount of the resource depends on the size of transaction. As a result, the authors of OVN estimate that their framework can only reasonably accommodate about 50 voters [25], whereas Ramachandran and Kantarcioglu limit their proposal to 100 voters [33].

We conclude that accounting for all the special cases that might occur is far from being trivial. The whole idea of using block chain as a ledger is to make its consistency independently verifiable. The definition of necessary and sufficient conditions for independent consistency verification must form an integral part of any proposal for a block chain based voting system.

3.4 Transaction registration issues

A common problem of public permissionless ledgers like Bitcoin and Ethereum is that the blocks in the chain are limited both in size and frequency, leading to a very low amount of transactions per second that the ledger can process (e.g.

[11] It is interesting to note that even though Agora is claimed to have an elaborate consistency verification mechanism, none of it was used in the Agora Sierra Leone event, where even the most basic block chain explorer tool was not provided for auditing `http://en.rfi.fr/africa/20180319-sierra-leones-electoral-commission-distances-itself-use-blockchain-during-polls`.

7 in case of Bitcoin and 15 in case of Ethereum)[12]. As this resource is shared worldwide, committing one transaction per vote to such a ledger is not realistic even for moderately-sized elections.

For electronic voting we need to provide voters and election officials with precise upper bounds on the transaction confirmation time. For the sake of this paper we assume that transaction is confirmed if it is included in any valid block, although more realistic requirement would be that several blocks (say, 6 in case of Bitcoin) extend the block containing the transaction. The time required to mine a new block in Bitcoin is 10 minutes on average, but the worst-case time can be much longer e.g. on the 1st of April, 2018, time between blocks number 516036 and 516037 is roughly 54 minutes.

Block generation time follows roughly the exponential distribution (although folklore, Bowden *et al.* [9] argue that this is not precisely correct if taken into account changing proof of work difficulty and network delays) where the cumulative distribution function $F(x) = 1 - e^{-x/\lambda}$ expresses the probability that a block is generated in x minutes, given the average rate of block generation λ (for Bitcoin $\lambda = 10$).

This allows us to make some rough estimates of block generation times. Probability that block generation takes more than 10 minutes is $1 - F(10) = e^{-1} \approx 0.37$, i.e. roughly third of the blocks take more than 10 minutes to generate. Probability that a block generation takes more than 50 minutes is $1 - F(10) = e^{-5} \approx 0.007$, hence we would expect to see 2 or 3 such blocks per day meaning that voters would have to wait more than 50 minutes to have any assurance about their vote actually being recorded.

Another serious drawback of public permissionless block chains is that the success of actually accepting a transaction into the ledger depends on financial incentives rather than the legislative need to create an immutable audit trail.

Garay *et al.* [14] prove the *liveness* property for Bitcoin block chain which informally says that if a transaction is broadcasted to honest nodes for a certain number of consecutive rounds, then the transaction will eventually be included in the block chain. However, this property relies on the assumption of synchronous network (see Pass et al. [31] for asynchronous networks) and that majority of hashing power is controlled by honest miners.

In particular, the last assumption is not quite true in real life – most miners behave rationally and will give preference to transactions with the highest fees. Transaction with no or little transaction fees might get completely neglected. As a result, successful data commitment can not be guaranteed in these types of block chains.

The main approach to speed up the process is to increase transaction rewards for the miners, rising the overall cost of the elections (but still not achieving a 100% guarantee due to the e.g. block-size limits). Note that transaction issuer will find out that the fee he offered for to the miners was too low only after some time has passed.

[12] `https://github.com/ethereum/wiki/wiki/Sharding-FAQ`

This problem is discussed most deeply by Noizat [29] who proposes various methods including using relatively high transaction fees to increase transaction priority, and election organizer becoming one of the miners. Even with using these methods, transaction confirmation may take several days [29,7,8]. Noizat and Bogucki [29,8] argue that this may be fine for at least some types of elections, but in our opinion, such a restriction limits applicability of block chain based voting remarkably.

The result of transaction confirmation being delayed (or even "forgotten" if it stays in the pool of pending transactions for too long) is devastating. The publicly verifiable audit trail will have blocks missing or occurring in the ledger in a wrong order. This will further complicate the consistency verification logic (see Section 3.3).

3.5 Performance issues

In order to incentivise block chain node contributions to the chain creation, transactions cost (crypto)money. On the other hand, cryptocurrencies like Bitcoin are very volatile. For example, in their paper published in April 2017, Bistarelli *et al.* [7] used the estimate 1Ƀ=547€ to compute the overall cost of running elections. However, by December 2017 the value of Bitcoin had reached over 16000€, making voting over 30 times more expensive. It is large monetary risk for election organiser. In those protocols, where the voter must initiate the transaction it effectively introduces a fee for voting.

Another performance problem specific to Bitcoin comes from the fact that it was designed primarily for monetary transactions and its ledger's ability to accommodate other types of data is rather limited. The most well known method, is to use the `OP_RETURN` field for free-form input, and the length of that field is only 80 bytes. This in turn means that instead of full data blobs, only their hashes can be committed to the Bitcoin block chains. If this is done carelessly, it can lead to new vulnerabilities.

A good example of this type of misdesign can be observed in the scheme proposed by Wu [41]. The scheme relies on ring signatures, but they can unfortunately be rather large. So, instead of signature σ, the hash $h(\sigma)$ is committed to the Bitcoin block chain. The original σ is kept by the Election Authority without any integrity protection. This means that an independent auditor has no way of resolving the dispute between a voter and the Election Authority if the latter e.g. claims that it has not received the signature σ from the voter in the first place. A potential solution is using a local block chain to store all the data required for auditing, and only committing snapshot hashes of the local block chain to the Bitcoin ledger. Such a solution is deployed e.g. by Agora and VoteWatcher [13,30].

Additional options for storing arbitrary data in Bitcoin block chain have been proposed in [36]. It is possible to store nearly 100kB in a single transaction. Content insertion services that fragment data over multiple transactions have been used to store files as large as 310.72kB in Bitcoin block chain [24]. It is, however, worth noting that usage of the `OP_RETURN` field or any other mechanism for storing non-currency data is discouraged by the Bitcoin development team.[13]

General purpose block chains (e.g. Ethereum) are better suited for arbitrary data storage. In principle, the storage available to Ethereum smart contracts is limited to 2^{256} words of 32 bytes, but it must be noted that any write operation requires 20k gas per 32 byte word [40].

The block chain storage is more general issue. A full copy of Bitcoin ledger has surpassed 150 GB [13], which makes it unreasonable to store it at every node. Also, due to objectionable content being committed to the Bitcoin block chain, storing certain parts of it can even be considered illegal [24].

As a possible solution to transaction cost, latency and storage problems, Agora uses several layers of block chains. On a lower level, a dedicated ledger is run, and only periodic aggregated snapshots are then committed to the Bitcoin block chain [13]. A similar approach is deployed by VoteWatcher. This can indeed reduce the Bitcoin transaction costs, but such an architecture complicates auditing, so essentially we get a trade-off between direct cost and complexity. This approach does not solve any transaction registration issues, meaning that some snapshots will be published in average time, some snapshots may take hours to publish and there are no guarantees.

3.6 Centralization of mining power

Even though one of the main targeted properties of permissionless public block chains is decentralization, this property does not necessarily hold in practice. For example, the most popular block chain implementations Bitcoin and Ethereum (unintentionally) incentivize the miners to group into centrally managed mining pools to allow for a constant stream of rewards to the pool participants. Additionally, being connected into geographically close nodes allows for faster broadcast of the mined blocks into pool members and thus gives an advantage in starting mining a new block earlier than other pools could [27].

Even more, the hash rate distribution within the pools is highly concentrated to a few nodes. In [27], Miller *et al.* have found that in 2015, only 2% of the Bitcoin mining nodes held three quarters of the mining power.

The state of centralization has not improved significantly since then. In a recent work [15], Gencer *et al.* have shown that in Bitcoin, four of the biggest pools hold 53% of average mining power. Ethereum shows a similar tendency towards centralization – three top Ethereum pools hold 61% of the mining power.

Centralization introduces a clear political risk. Several estimates indicate that the top miners are exclusively located in China, with around 80% of mining power of the Bitcoin network belonging to Chinese pools[14][15]. This makes it in principle possible to influence majority of the miners from one central political

[13] `https://bitcoin.org/en/release/v0.9.0#opreturn-and-data-in-the-block-chain`

[14] `https://www.buybitcoinworldwide.com/mining/pools/`

[15] `https://altcointoday.com/ethereum-mining-hashrate-distribution-issues/`

authority, thus violating the assumptions required for immutability of the block chain [28]. This might not be a serious concern for smaller elections (say, electing a rector of a university), but is clearly a worry for governmental elections.

4 Trade-offs

Block chain is not a miracle solution for all the voting-related problems. While it has strong integrity properties definitely required by election systems, a block chain ledger is non-trivial to set up and interface with other components. Several trade-offs need to be made and the interplay of these trade-offs may have unwanted consequences.

4.1 Expressive power *vs.* complexity

The first trade-off one needs to consider is between the expressiveness of the claims the block chain commitments have, versus the complexity that one needs to accept while verifying them.

For example, we may want to automate exception handling to the level where the correctness of possible error fixes is certified by smart contracts, but this implies the need for complicated certification logic which is accompanied by performance penalties of running the smart contracts.

Alternatively, we may want to make use of commitments to a public ledger like Bitcoin to utilize the whole power of global trust. However, due to Bitcoin's limited capacity of handling external inputs, we need an extended mechanism of integrity certification (essentially, another layer of block chain ledger below Bitcoin). This in turn complicates the verification logic.

Smart contracts have more potential in terms of expressive power – there is support for arbitrary data structures and the validation logic itself is published to the chain and executed by the nodes. Of course, this has severe implications on performance.

It is a question if in case of general purpose ledgers it is enough to verify the consistency of the ledger with respect to one particular election or should the consistency of the ledger as a whole also be assured, so that the verification of full nodes becomes part of the election audit procedure.

4.2 Small scale *vs.* large scale

There are issues that are potentially easier to solve on small-scale elections.

For example, eligibility verification is much easier when everyone knows everyone else, but this can only be the case in very localized settings. For larger-scale events we have to accept reliance on an external identity provider who may be malicious without detection.

The OVN smart contract [25] is a great example how to ensure ballot secrecy in boardroom elections where we can somehow assure that everybody is going to participate in the whole process. The protocol is clearly not usable for large-scale elections.

Also, several techniques potentially useful for block chain voting (e.g. ring signatures) have a significant performance penalty. For large-scale elections one has to avoid them, also losing the benefits they provide.

4.3 Trust *vs.* cost

Using Bitcoin ledger for commitments is very appealing because of a lot of public trust in the integrity properties it provides and a large community relying on this trust already. However, the cost of Bitcoin transactions (and also transactions in other public ledgers) has been very volatile in the near past. For example, on December 22nd 2017, Bitcoin transaction fee spiked to $55.16.[16]

This means that the election organizers must have a lot of flexibility in budgeting. However, public election authorities tend to operate under budget constraints and prefer well-predictable costs. While one of the targets of block chain voting is reducing the overall price tag of elections [29,26,7], this effect may be reduced by the potential volatility of the costs.

Using local block chains mitigates this problem considerably, but this solution also deprives us of the benefit of public trust. Also, the costs of setting up and running a local block chain are non-negligible, although better-predictable.

4.4 Usability *vs.* individual verifiability

In both academic and industrial communities a general consensus is that the electronic voting schemes have to provide some kind of verifiability. The individual aspect of the verifiability, as defined in [22], should convince the voter that the vote has been stored correctly by the election provider.

To be able to provide individual verifiability, the election provider needs to store the ballot and construct a receipt which can be used for verifying the correct storage. The receipt could contain storage location identifier, proof of registering the ballot [18], block chain block identifier etc.

In case block chain is used for storing any aspect of the ballot, then independently of the underlying block chain technology used, the voter needs to wait until the information has been stored in the block chain. However, the latency of block chain storage can be rather long and the variance can be large. In extreme cases, the storage confirmation may even not arrive in reasonable time (see Section 3.4).

From the voter's perspective, this means that either receiving the receipt or verifying the ballot may take considerable time. The voter would then need to return later to complete verification and this decreases usability of the whole scheme. As a result, the number of vote verifications is expected to drop, together with the overall public confidence in election integrity.

[16] `https://bitinfocharts.com/comparison/bitcoin-transactionfees.html`

5 Conclusions

Even though applying block chain as an integrity assurance measure may seem straightforward for electronic voting, extra assumptions and trade-offs required make setting up such a system a non-trivial task.

Many of the proposals that we have considered try to make use of public permissionless economically incentivised block chains, mostly either Bitcoin or Ethereum. On one hand this makes a lot of sense, since they are widely used and trusted. However, these ledgers have major drawbacks like the tendency for centralisation, providing no guarantees of transaction acceptance and performance limitations. In our opinion, due to these drawbacks such block chains have very limited use for electronic voting. To be considered useful for voting, the block chain must accept authorized commitments immediately and unconditionally.

None of the proposals we studied had a complete description of conditions that need to be verified in order for the voting event to be considered right. Currently, using smart contracts seems to be the most systematic approach to deal with this issue, but systems using smart contracts so far imply a significant performance penalty, strongly limiting e.g. the number of voters.

Also, majority of the proposals ignored the need for exception handling. We conjecture that full consistency verification of block chain based voting systems is rather complex, defying the original target of transparency. It may be the case that simplicity of the verification routines needs to be recognised as a development requirement of its own right.

We would like to conclude the paper by citing Josh Benaloh [32]:

> *I find myself debunking a blockchain voting effort about every few weeks. It feels like a very good fit for voting, until you dig a couple millimeters below the surface.*

Even though such a statement may be a bit too categorical, we agree that in all of the proposals we considered, many of the shortcomings and trade-offs of block chains were addressed insufficiently. Considerably deeper research is required to settle a good design for a block chain based electronic voting system.

Acknowledgements

The research leading to these results has received funding from the Estonian Research Council under Institutional Research Grant IUT27-1, European Union's Horizon 2020 research and innovation programme under grant agreement No 780477 (project PRIViLEDGE), and the European Regional Development Fund through the Estonian Centre of Excellence in ICT Research (EXCITE) and the grant number EU48684.

References

1. Online Voting. Successfully Solving the Challenges. TIVI Whitepaper. `http://www.smartmatic.com/fileadmin/user_upload/Whitepaper_Online_Voting_Challenge_Considerations_TIVI.pdf`.

2. The Social Smart Contract, 2018. `http://paper.democracy.earth/`.
3. Roman Alyoshkin. Polys online voting system. Whitepaper. `https://polys.me/assets/docs/Polys_whitepaper.pdf`.
4. Dave Bayer, Stuart Haber, and W Scott Stornetta. Improving the efficiency and reliability of digital time-stamping. In *Sequences II*, pages 329–334. Springer, 1993.
5. Susan Bell, Josh Benaloh, Michael D Byrne, Dana DeBeauvoir, Bryce Eakin, Gail Fisher, Philip Kortum, Neal McBurnett, Julian Montoya, Michelle Parker, et al. STAR-Vote: A secure, transparent, auditable, and reliable voting system. *USENIX Journal of Election Technology and Systems (JETS)*, 1(1):18–37, 2013.
6. Josh Benaloh and Eric Lazarus. The trash attack: An attack on verifiable voting systems and a simple mitigation. Technical report, 2011. MSR-TR-2011-115, Microsoft.
7. Stefano Bistarelli, Marco Mantilacci, Paolo Santancini, and Francesco Santini. An End-to-end Voting-system Based on Bitcoin. In *Proceedings of the Symposium on Applied Computing*, SAC '17, pages 1836–1841. ACM, 2017.
8. Brianna Bogucki. Buying Votes in the 21st Century: The Potential Use of Bitcoins and Blockchain Technology in Electronic Voting Reform. *Asper Review of International Business and Trade Law*, 17:59–84, 2017.
9. Rory Bowden, Holger Paul Keeler, Anthony E. Krzesinski, and Peter G. Taylor. Block arrivals in the Bitcoin blockchain. *CoRR*, abs/1801.07447, 2018.
10. Nikos Chondros, Bingsheng Zhang, Thomas Zacharias, Panos Diamantopoulos, Stathis Maneas, Christos Patsonakis, Alex Delis, Aggelos Kiayias, and Mema Roussopoulos. D-DEMOS: A distributed, end-to-end verifiable, internet voting system. In *ICDCS 2016*, pages 711–720. IEEE Computer Society, 2016.
11. Chris Culnane and Steve A. Schneider. A peered bulletin board for robust use in verifiable voting systems. In *IEEE CSF 2014*, pages 169–183. IEEE Computer Society, 2014.
12. Nazim Faour. Transparent Voting Platform Based on Permissioned Blockchain. Master's thesis, Higher School of Economics, National Research University, Russia, 2018. `https://arxiv.org/abs/1802.10134`.
13. Leonardo Gammar, Bryan Ford, and Jaron Lukasiewicz. Agora. Bringing our voting systems into the 21st century. Whitepaper, version 0.1. `https://agora.vote/Agora_Whitepaper_v0.1.pdf`.
14. Juan Garay, Aggelos Kiayias, and Nikos Leonardos. The bitcoin backbone protocol: Analysis and applications. In *EUROCRYPT 2015: Advances in Cryptology – EUROCRYPT 2015*, volume 9057 of *LNCS*, pages 281–310. Springer, 2015.
15. Adem Efe Gencer, Soumya Basu, Ittay Eyal, Robbert van Renesse, and Emin Gün Sirer. Decentralization in Bitcoin and Ethereum Networks, 2018. `https://arxiv.org/abs/1801.03998`.
16. Stuart Haber and W. Scott Stornetta. How to Time-Stamp a Digital Document. In Alfred J. Menezes and Scott A. Vanstone, editors, *Advances in Cryptology-CRYPT0' 90*, volume 537 of *LNCS*, pages 437–455. Springer Berlin Heidelberg, 1991.
17. Feng Hao, Peter Y. A. Ryan, and Piotr Zielinski. Anonymous voting by two-round public discussion. *IET Information Security*, 4(2):62–67, 2010.
18. Sven Heiberg, Tarvi Martens, Priit Vinkel, and Jan Willemson. Improving the Verifiability of the Estonian Internet Voting Scheme. In Robert Krimmer, Melanie Volkamer, Jordi Barrat, Josh Benaloh, Nicole Goodman, Peter Y. A. Ryan, and Vanessa Teague, editors, *E-Vote-ID 2016: Electronic Voting*, volume 10141 of *LNCS*, pages 92–107. Springer, 2017.

19. Aggelos Kiayias, Annabell Kuldmaa, Helger Lipmaa, Janno Siim, and Thomas Zacharias. On the security properties of e-voting bulletin boards. In *Security and Cryptography for Networks - 11th International Conference, SCN 2018, Amalfi, Italy, September 5 - September 7, 2018, Proceedings*, 2018. To appear.
20. Kevin Kirby, Anthony Masi, and Fernando Maymi. Votebook. A proposal for a blockchain-based electronic voting system, September 2016. http://www.economist.com/sites/default/files/nyu.pdf.
21. Ali Kaan Koç, Emre Yavuz, Umut Can Çabuk, and Gökhan Dalkiliç. Towards Secure E-Voting Using Ethereum Blockchain, 2018. 6th International International Symposium on Digital Forensic and Security (ISDFS), Antalya, Turkey. https://www.researchgate.net/publication/323318041.
22. Steve Kremer, Mark Ryan, and Ben Smyth. Election verifiability in electronic voting protocols. In *Computer Security - ESORICS 2010, 15th European Symposium on Research in Computer Security, Athens, Greece, September 20-22, 2010. Proceedings*, pages 389–404, 2010.
23. Kibin Lee, Joshua I James, Tekachew Gobena Ejeta, and Hyoung Joong Kim. Electronic voting service using block-chain. *The Journal of Digital Forensics, Security and Law: JDFSL*, 11(2):123–135, 2016. https://commons.erau.edu/jdfsl/vol11/iss2/8/.
24. Roman Matzutt, Jens Hiller, Martin Henze, Jan Henrik Ziegeldorf, Dirk Müllmann, Oliver Hohlfeld, and Klaus Wehrle. A Quantitative Analysis of the Impact of Arbitrary Blockchain Content on Bitcoin. In *Proceedings of the 22nd International Conference on Financial Cryptography and Data Security (FC). Springer*, 2018. http://fc18.ifca.ai/preproceedings/6.pdf.
25. Patrick McCorry, Siamak F. Shahandashti, and Feng Hao. A Smart Contract for Boardroom Voting with Maximum Voter Privacy. In Aggelos Kiayias, editor, *Financial Cryptography and Data Security*, volume 10322 of *LNCS*, pages 357–375. Springer International Publishing, 2017.
26. Christian Meter. Design of Distributed Voting Systems. Master's thesis, Heinrich-Heine-Universität Düsseldorf, 2015. https://arxiv.org/pdf/1702.02566.pdf.
27. Andrew Miller, James Litton, Andrew Pachulski, Neal Gupta, Dave Levin, Neil Spring, and Bobby Bhattacharjee. Discovering bitcoin's public topology and influential nodes, May 2015. http://cs.umd.edu/projects/coinscope/coinscope.pdf.
28. Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008. https://bitcoin.org/bitcoin.pdf.
29. Pierre Noizat. Chapter 22 – blockchain electronic vote. In David Lee Kuo Chuen, editor, *Handbook of Digital Currency*, pages 453 – 461. Academic Press, San Diego, 2015.
30. Ryan Osgood. The Future of Democracy: Blockchain Voting. *COMP116: Information Security*, 2016. http://www.cs.tufts.edu/comp/116/archive/fall2016/rosgood.pdf.
31. Rafael Pass, Lior Seeman, and abhi shelat. Analysis of the Blockchain Protocol in Asynchronous Networks. In *Advances in Cryptology – EUROCRYPT 2017*, volume 10211 of *LNCS*, pages 643–673. Springer, 2017.
32. Morgen E. Peck. Blockchain world – Do you need a blockchain? This chart will tell you if the technology can solve your problem. *IEEE Spectrum*, 54(10):38–60, October 2017.
33. Aravind Ramachandran and Murat Kantarcioglu. Using Blockchain and smart contracts for secure data provenance management, 2017. arXiv preprint arXiv:1709.10000, https://arxiv.org/abs/1709.10000.

34. Daniel Sandler, Kyle Derr, and Dan S Wallach. VoteBox: A Tamper-evident, Verifiable Electronic Voting System. In *17th USENIX Security Symposium*, pages 349–360, 2008.
35. Daniel Sandler and Dan S Wallach. Casting Votes in the Auditorium. In *USENIX/ACCURATE Electronic Voting Technology Workshop*, 2007.
36. Andrew Sward, Vecna OP_0, and Forrest Stonedahl. Data Insertion in Bitcoin's Blockchain, 2017. Computer Science: Faculty Scholarship & Creative Works. `https://digitalcommons.augustana.edu/cscfaculty/1/`.
37. Nick Szabo. Smart contracts: building blocks for digital markets. *EXTROPY: The Journal of Transhumanist Thought, (16)*, 1996.
38. Nick Szabo. Formalizing and Securing Relationships on Public Networks. *First Monday*, 2(9), 1997.
39. Yu Takabatake, Daisuke Kotani, and Yasuo Okabe. An anonymous distributed electronic voting system using Zerocoin. *IEICE Technical Report*, 116(282), 11 2016. `http://hdl.handle.net/2433/217329`.
40. Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger EIP-150 REVISION (759dccd - 2017-08-07), 2017. Accessed: 2018-01-03.
41. Yifan Wu. An E-voting System based on Blockchain and Ring Signature. Master's thesis, University of Birmingham, 2017. `https://www.dgalindo.es/mscprojects/yifan.pdf`.
42. Zhichao Zhao and T.-H. Hubert Chan. How to Vote Privately Using Bitcoin. In Sihan Qing, Eiji Okamoto, Kwangjo Kim, and Dongmei Liu, editors, *Information and Communications Security: 17th International Conference, ICICS 2015, Beijing, China, December 9-11, 2015, Revised Selected Papers*, volume 9543 of *LNCS*, pages 82–96. Springer, 2016.

Risk-Limiting Audits

Ballot-polling Risk Limiting Audits for IRV Elections

Michelle Blom, Peter J. Stuckey, and Vanessa J. Teague

[michelle.blom,p.stuckey,vjteague]@unimelb.edu.au
School of Computing and Information Systems
The University of Melbourne
Parkville, Australia

Abstract. Risk-limiting post election audits guarantee a high probability of correcting incorrect election results, independent of why the result was incorrect. Ballot-polling audits select ballots at random and interpret those ballots as evidence for and against the actual recorded result, continuing this process until either they support the recorded result, or they fall back to a full manual recount. Ballot-polling for first-past-the-post elections is well understood, and used in some US elections. We define a number of approaches to ballot-polling risk-limiting audits for Instant Runoff Voting (IRV) elections. We show that for almost all real elections we found, we can perform a risk-limiting audit by looking at only a small fraction of the total ballots (assuming no errors).

1 Introduction

Instant Runoff Voting (IRV) is a system of preferential voting in which voters rank candidates in order of preference. IRV is used for all parliamentary lower house elections in Australia, parliamentary elections in Fiji and Papua New Guinea, presidential elections in Ireland and Bosnia/Herzogovinia, and local elections in numerous locations worldwide, including the UK and United States. Given candidates c_1, c_2, c_3, and c_4, each vote in an IRV election is a (*possibly partial*) ranking of these candidates. A vote with the ranking $[c_1, c_2, c_3]$ expresses a first preference for candidate c_1, a second preference for c_2, and a third for c_3. The tallying of votes proceeds by distributing each vote to its first ranked candidate. The candidate with the smallest number of votes is eliminated, with their votes redistributed to subsequent, less preferred candidates. Elimination proceeds in this fashion, until a single candidate w remains, who is declared the winner.

Risk Limiting Audits [6] (RLAs) provide strong statistical evidence that the reported outcome of an election is correct, or revert to a manual recount if it is wrong. The probability that the audit fails to detect a wrong outcome is bounded by a *risk limit*. An RLA with a risk limit of 1%, for example, has at most a 1% chance of failing to detect that a reported election outcome is wrong. In this paper we present several methods for undertaking ballot-polling RLAs of IRV elections, by adapting a ballot-polling RLA method (BRAVO) designed for first-past-the-post or k-winner plurality elections [7].

Blom *et al* [3] demonstrated an efficient algorithm for exact IRV margin computation. This immediately allows for a risk-limiting *comparison audit* [6], assuming that there is infrastructure for comparing ballots with their electronic record. This would consist of simply assessing the number of discrepancies until the hypothesis that there were enough to change the outcome could be rejected. However, that might be very inefficient because it counts every error equally, including those that help the apparent winner or rearrange candidates with no hope of winning. It might be possible to extend Stark's sharper discrepancy measure [10] to IRV, but this is challenging because it may be hard to compute the implications of a particular discrepancy.

In this paper we instead consider ballot-polling audits for IRV, by applying BRAVO to auditing certain facts about an IRV election. In a k-winner plurality contest, BRAVO maintains a running statistic T_{wl} for each pair of apparent winner w and loser l. These statistics are updated as ballots are drawn uniformly at random. A ballot that shows a valid vote for winner w increases the T_{wl} statistic (by an amount dependent on the reported votes for the two candidates), while a ballot showing a valid vote for the loser l decreases it. When each statistic exceeds a threshold, dependent on the risk limit, we know that we have seen enough evidence to reject the hypothesis that l beat w.

Each round of IRV elimination could be regarded as a multiple-winner plurality election—this idea was explored in [9]. We denote this by **IRV**, annotated with the round and eliminated candidates. Adapting BRAVO directly to this is described in Section 5.1. This is sound, but wastes a lot of auditing work proving a much stronger result than necessary—the elimination order may be wrong though the final outcome is correct. One optimization is to eliminate batches of low-tally candidates at once when this provably doesn't affect the final outcome. These batch eliminations can also be easily audited with BRAVO—this is described in Section 5.2.

An even simpler fact turns out to be very powerful: suppose we wish to reject the hypothesis that w was eliminated before l. We can apply BRAVO immediately, counting every ballot with a *first preference* for w as a vote for w, which is conservative because w must have *at least* this tally at every stage. Any vote that mentions l without a higher preference for w is attributed to l, which is also conservative because l can have *at most* this tally. If BRAVO rejects the hypothesis that l can beat w, then we can reject the hypothesis that w is eliminated before l. We call this the Winner Only hypothesis, denoted **WO**(l, w). It can also be conditioned on a set of already-eliminated candidates $\mathcal{C}$—preferences for those candidates are simply ignored when auditing the w-l pair.

Winner-only audits are described in Section 5.3. A surprising result of this paper is that WO alone often suffices for an efficient, complete audit. In about half the real elections we simulated auditing, we found that for the announced winner w, for every loser l, hypothesis **WO**(l, w) could be efficiently rejected using BRAVO. This confirms that w won, while sidestepping almost all the complexity of IRV.

The key contribution of this paper is a good heuristic for choosing which combination of facts to audit, using BRAVO, in order to provide an efficient risk-limiting audit of an IRV election result. We present an algorithm, denoted *audit-irv*, that finds a sufficient set of facts (e.g., some version of **IRV** or **WO**(c_1**,** c_2) given that c_3 and c_4 have been eliminated) to prove that w won. All of these facts can be audited simultaneously using BRAVO. If one of the necessary facts is false, this will be detected, with probability of at least $1 - \alpha$, by the BRAVO audit at risk limit α.

Ideally we would like to ensure that *audit-irv* selects the set of facts that produce an optimally efficient audit, but this is very difficult. When BRAVO is assessing only a single winner, its average sample number (ASN) can be easily computed, but the expected number of samples for eliminating multiple (perhaps related) hypotheses can (as far as we know) be assessed only by simulation. *audit-irv* selects the collection of facts that minimizes the maximum ASN for each fact taken separately—this is what we mean by the "optimal" auditing program below. However, this may not actually be an optimally efficient audit, or even the optimal application of BRAVO, because it is possible that some other combination of facts can be checked together more efficiently.

Our simulations show that *audit-irv* plans a feasible IRV audit, using BRAVO, for almost all the real IRV elections we could find. Although some still require large audits, this is probably inevitable because their margins are small.

Definitions and background are in Section 3. Section 4 introduces the BRAVO ballot-polling RLA for first-past-the-post elections. Section 5 describes our ballot-polling approaches, then Section 6 simulates and evaluates them on a suite of IRV instances.

2 Related Work

There is a growing literature on the use of risk-limiting audits for auditing the outcome of varying types of election [7, 9]. Risk-limiting audits have been applied to a number of plurality (first-past-the-post) elections, including four 2008 elections in California [4] and elections in over 50 Colorado counties in 2017. General auditing procedures designed to enhance electoral integrity have been outlined by [1]. The BRAVO ballot-polling risk-limiting audit of [7], designed for first-past-the-post elections, forms the basis of our IRV ballot-polling audits.

Several approaches for designing a risk-limiting comparison audit of an IRV election have been proposed [9]. Such audits retrieve paper ballots and compare them to their corresponding electronic record – an erroneous ballot is one that does not match its electronic record. The first of these methods determines whether replacing an erroneous ballot with its correct representation changes the margin of victory of the election. The second is based on auditing the elimination order, performing a plurality audit for each round of counting. The audit performed at round r checks whether the set of candidates eliminated prior to r, viewed as a single 'super candidate', loses to the set of remaining candidates (that are still standing). We consider a similar approach, in the context of a ballot-polling audit, in this paper. We show, however, that we can more efficiently audit an IRV election outcome by simply verifying that the reported winner was not defeated by any other candidate. The third method proposed by [9] samples K ballots, and determines whether the number of erroneous ballots exceeds a defined threshold, based on the margin of victory of the election.

In parliamentary elections, such as Australian state and federal elections, the overall outcome is determined by the results of a set of such elections, one for each of a set of regions or districts. In the context of multi-level elections such as these, [5] present a linear programming-based method to compute the statistical confidence with which each district-level election should be audited, given an appropriate risk-limiting auditing method, while minimising the expected number of ballots that must be checked overall. Their approach ensures that the overall outcome is audited to a given level of statistical confidence, while varying the extent to which each district-level election is audited.

For a risk-limiting audit, the margin of victory of the election provides an indication of how many ballots will need to be sampled. Automatic methods for computing electoral margins for IRV elections have been presented by [8, 3, 2].

```
Initially, all candidates remain standing (are not eliminated)
While there is more than one candidate standing
    For every candidate $c$ standing
        Tally (count) the ballots in which $c$ is the highest-ranked
        candidate of those standing
    Eliminate the candidate with the smallest tally
The winner is the one candidate not eliminated
```

Fig. 1: An informal definition of the IRV counting algorithm.

3 Preliminaries

In a first-past-the-post (FPTP) election, a voter marks a single candidate on their ballot when casting their vote. The candidate who receives the most votes is declared the winner. The BRAVO risk limiting audits of [7] are designed for k-winner FPTP contests. A voter may vote for up to k of the candidates on their ballot, and the k candidates with the highest number of votes are declared winners. IRV, in contrast, is a form of preferential voting in which voters express a preference ordering over a set of candidates on their ballot. The tallying of votes in an IRV election proceeds by a series of rounds in which the candidate with the lowest number of votes is eliminated (see Figure 1) with the last remaining candidate declared the winner. All ballots in an eliminated candidate's tally are distributed to the next most-preferred (remaining) candidate in their ranking.

Let $\mathcal{C}$ be the set of candidates in an IRV election $\mathcal{B}$. We refer to sequences of candidates π in list notation (e.g., $\pi = [c_1, c_2, c_3, c_4]$), and use such sequences to represent both votes and elimination orders. An election $\mathcal{B}$ is defined as a multiset[1] of ballots, each ballot $b \in \mathcal{B}$ a sequence of candidates in $\mathcal{C}$, with no duplicates, listed in order of preference (most preferred to least preferred). Throughout this paper we use the notation $\mathit{first}(\pi) = \pi(1)$ to denote the first candidate in a sequence π. In each round of vote counting, there are a current set of eliminated candidates $\mathcal{E}$ and a current set of candidates still standing $\mathcal{S} = \mathcal{C} \setminus \mathcal{E}$. The winner c_w is the last standing candidate.

Definition 1. ***Projection*** $\mathbf{p}_{\mathcal{S}}(\pi)$ *We define the projection of a sequence π onto a set $\mathcal{S}$ as the largest subsequence of π that contains only elements of $\mathcal{S}$. (The elements keep their relative order in π). For example:*
$p_{\{c_2,c_3\}}([c_1, c_2, c_4, c_3]) = [c_2, c_3]$ *and* $p_{\{c_2,c_3,c_4,c_5\}}([c_6, c_4, c_7, c_2, c_1]) = [c_4, c_2]$.

Each candidate $c \in \mathcal{C}$ has a *tally* of ballots. Ballots are added to this tally upon the elimination of a candidate $c' \in \mathcal{C} \setminus c$, and are redistributed upon the elimination of c.

Definition 2. ***Tally*** $\mathbf{t}_{\mathcal{S}}(\mathbf{c})$ *Given candidates $\mathcal{S} \subseteq \mathcal{C}$ are still standing in an election $\mathcal{B}$, the tally for a candidate $c \in \mathcal{C}$, denoted $t_{\mathcal{S}}(c)$, is defined as the number of ballots $b \in \mathcal{B}$ for which c is the most-preferred candidate of those remaining. Recall that $p_{\mathcal{S}}(b)$ denotes the sequence of candidates mentioned in b that are also in $\mathcal{S}$.*

$$t_{\mathcal{S}}(c) = \ | \, [b \mid b \in \mathcal{B}, c = \mathit{first}(p_{\mathcal{S}}(b))] \, | \tag{1}$$

[1] A multiset allows for the inclusion of duplicate items.

Ranking	Count
$[c_2, c_3]$	4000
$[c_1]$	20000
$[c_3, c_4]$	9000
$[c_2, c_3, c_4]$	6000
$[c_4, c_1, c_2]$	15000
$[c_1, c_3]$	6000

(a)

Candidate	Rnd1	Rnd2	Rnd3
c_1	26000	26000	26000
c_2	10000	10000	—
c_3	9000	—	—
c_4	15000	24000	30000

(b)

Table 1: An example IRV election, stating (a) the number of ballots cast with each listed ranking over four candidates, and (b) the tallies after each round of counting.

The *primary vote* of candidate $c \in \mathcal{C}$, denoted $f(c)$, is the number of votes $b \in \mathcal{B}$ for which c is ranked highest. Note that $f(c) = t_{\mathcal{C}}(c)$.

$$f(c) = \ | \ [b \mid b \in \mathcal{B}, c = \mathit{first}(b)] \ | \qquad (2)$$

Example 1. Consider the IRV election of Table 1. The tallies of c_1, c_2, c_3, and c_4, in the 1^{st} counting round are 26000, 10000, 9000, and 15000 votes. Candidate c_3 is eliminated, and 9000 ballots are distributed to c_4, who now has a tally of 24000. Candidate c_2, on 10000 votes, is eliminated next with 6000 of their ballots given to c_4 (the remainder have no subsequent preferences and are exhausted). Candidates c_1 and c_4 remain with tallies of 26000 and 30000. Candidate c_1 is eliminated and c_4 elected. □

4 Ballot-polling risk-limiting audits for FPTP

The aim of ballot-polling risk limiting audits is to be reassured that the results of the election are valid even if some counting errors occurred. To this end we will consider two versions of the statistics defined in the previous section. We use the regular definition for the *recorded* values made during the election, and add a tilde ˜ to mean the *actual* values which should have been calculated. Hence $f(c)$ is the recorded primary vote for candidate c and $\tilde{f}(c)$ is the actual primary vote for the candidate.

For now we consider a simple k-winner from n candidates FPTP election where the k candidates who have the greatest number of votes are elected. All winners are elected simultaneously and there is no transfer of votes. Given a set of $\mathcal{C}$ candidates ($|\mathcal{C}| = n$) there will be a set of $\mathcal{W}$ *winners* ($|\mathcal{W}| = k$) and $\mathcal{L}$ *losers* ($|\mathcal{L}| = n - k$).

We now present the BRAVO algorithm [7] for ballot-polling risk-limiting audits of such elections (Figure 2(a)). BRAVO is applicable in elections where each ballot may express a vote for one or more candidates. For our proposed IRV audits, we apply BRAVO in contexts where each ballot represents a vote for a single candidate only (i.e., in any round of an IRV count, each ballot belongs to the tally of no more than one candidate). We describe the BRAVO algorithm in the context where each ballot b is equivalent to $\mathit{first}(b)$. Then $f(c)$ is the tally of votes for each candidate $c \in \mathcal{C}$.

The ballot-polling risk-limiting audit independently tests $k(n-k)$ null hypotheses $\{\tilde{f}(w) \leq \tilde{f}(l)\}$ for each winner/loser pair. A statistic for each test $\{T_{wl}\}$ is updated when a ballot is drawn for either its winner or its loser.

Given an overall risk limit α we can estimate for each hypothesis the number of ballot polls we expect will be required to reject the hypothesis assuming the election counts are perfectly accurate. Let p_c be the proportion of recorded votes for candidate c, i.e. $p_c = f(c)/|\mathcal{B}|$. Let s_{wl} be the proportion of recorded votes for the winner w of the votes for the winner and loser, $s_{wl} = p_w/(p_w + p_l)$. Clearly $s_{wl} > 0.5$. Then the *Average Sample Number (ASN)* [7], that is the expected number of samples to reject the null hypothesis $\{\tilde{p}_w \leq \tilde{p}_l\}$ assuming the recorded counts are correct, is given by:

$$ASN \simeq \frac{ln(1/\alpha) + 0.5ln(2s_{wl})}{(p_w ln(2s_{wl}) + p_l ln(2 - 2s_{wl}))} \tag{3}$$

Example 2. Consider the first round of the IRV election of Example 1. The null hypotheses we need to reject are $\tilde{f}(c_1) \leq \tilde{f}(c_3)$, $\tilde{f}(c_2) \leq \tilde{f}(c_3)$, $\tilde{f}(c_4) \leq \tilde{f}(c_3)$. We calculate $p_1 = 26000/60000$, $p_2 = 10000/60000$, $p_3 = 9000/60000$, $p_4 = 15000/60000$ and $s_{13} = 26000/35000$, $s_{23} = 10000/19000$, and $s_{43} = 15000/24000$. The ASN for rejecting each hypothesis, assuming $\alpha = 0.05$, is 44.5, 6885, and 246 respectively. □

5 Ballot-polling risk-limiting audits for IRV

5.1 Auditing a particular elimination order

The simplest approach to applying ballot-polling risk limiting auditing to IRV is to consider the IRV election as a number of simultaneous FPTP elections, one for each IRV round. This was previously suggested by Sarwate *et al* [9], although they do not explore it algorithmically. Note that this may perform much more auditing than required, since it verifies more than just that the eventual winner is the correct winner, but that every step in the IRV election was correct (with some confidence).

Given an election $\mathcal{B}$ of n candidates $\mathcal{C}$ let the computed elimination order of the candidates be $\pi = [c_1, c_2, \ldots, c_{n-1}, c_n]$ where c_1 is the first eliminated candidate, c_2 the second, etc, and c_n the eventual winner.

Each IRV round corresponds to a FPTP election. In the i^{th} round we have a FPTP election where $l = c_i$ is eliminated. The set of candidates of this election are $C_l = \{c_j \mid i \leq j \leq n\}$ with recorded tally $t_{C_l}(c)$ for each candidate $c \in C_l$, and loser $l = c_i$ and $n - i$ winners $C_l \setminus \{l\}$.

We can audit all these FPTP elections simultaneously, by simply considering all the null hypotheses that would violate the computed result. These are $\{\tilde{t}_{C_l}(c) \leq \tilde{t}_{C_l}(c_l) \mid 1 \leq i \leq n-1, l = c_i, c \in C_i \setminus \{l\}\}$. We represent these hypotheses by a pair (w, l) of winner $w = c$, and loser $l = c_i$. The statistic maintained for this test is T_{wl}. Note each loser only loses in one round so there is no ambiguity.

The algorithm is shown in Figure 2(b). The set of hypotheses H are again pairs (w, l) of winner w and loser l, but they are interpreted as a hypothesis for the FPTP election corresponding to the round where l was eliminated. This means the calculation of the expected ratio of votes s_{wl} must be made using the tallies from this round. It also means we must consider every ballot to see how it is interesting for that particular hypothesis. Note that for example a ballot that is exhausted after k rounds will not play any role in determining statistics for later round hypotheses.

```
bravo($\tilde{\mathcal{B}}$,$\mathcal{W}$,$\mathcal{L}$,$\alpha$,$M$)
  for($w \in \mathcal{W}, l \in \mathcal{L}$)
    $T_{wl}$ := 1
    $s_{wl}$ := $f(w)/(f(w) + f(l))$
  $H$ := $\mathcal{W} \times \mathcal{L}$
  $m$ := 0
  while($m < M \wedge H \neq \emptyset$)
    randomly draw ballot $b$ from $\tilde{\mathcal{B}}$
    $m$ := $m + 1$
    if($first(b) \in \mathcal{W}$)
      for($(w, l) \in H, w = first(b)$)
        $T_{wl}$ := $T_{wl} \times 2s_{wl}$
        if($T_{wl} \geq 1/\alpha$)
          % reject the null hypothesis
          $H = H - \{(w, l)\}$
    elseif($first(b) \in \mathcal{L}$)
      for($(w, l) \in H, l = first(b)$)
        $T_{wl}$ := $T_{wl} \times 2(1 - s_{wl})$
  if($H = \emptyset$)
    % reported results stand
    return $true$
  else % full recount required
    return $false$
```

(a)

```
irvbravo($\tilde{\mathcal{B}}$,$\pi$,$\alpha$,$M$)
  $H$ := $\emptyset$
  for($i \in 1..|\pi| - 1$)
    $l$ := $\pi(i)$
    $C_l$ := $\{\pi(i), \pi(i + 1), \ldots, \pi(|\pi|)\}$
    for($j \in i + 1..|\pi|$)
      $w$ := $\pi(j)$
      $T_{wl}$ := 1
      $s_{wl}$ := $t_{C_l}(w)/(t_{C_l}(w) + t_{C_l}(l))$
      $H$ := $H \cup \{(w, l)\}$
  $m$ := 0
  while($m < M \wedge H \neq \emptyset$)
    randomly draw ballot $b$ from $\tilde{\mathcal{B}}$
    $m$ := $m + 1$
    for($(w, l) \in H$)
      if($w = first(p_{C_l}(b))$)
        $T_{wl}$ := $T_{wl} \times 2s_{wl}$
        if($T_{wl} \geq 1/\alpha$)
          % reject the null hypothesis
          $H = H - \{(w, l)\}$
      elseif($l = first(p_{C_l}(b))$)
        $T_{wl}$ := $T_{wl} \times 2(1 - s_{wl})$
  if($H = \emptyset$)
    % reported results stand
    return $true$
  else % full recount required
    return $false$
```

(b)

Fig. 2: (a) BRAVO algorithm for a ballot-polling RLA audit of a FPTP election with actual ballots $\tilde{\mathcal{B}}$, declared winners $\mathcal{W}$, declared losers $\mathcal{L}$, risk limit α and limit on ballots checked M, and (b) algorithm for a ballot-polling RLA of an IRV election with actual ballots $\tilde{\mathcal{B}}$, order of elimination π, risk limit α and limit on ballots checked M. In both algorithms, ballots are drawn uniformly at random from $\tilde{\mathcal{B}}$.

Example 3. Consider the IRV election shown in Example 1. The null hypotheses we need to reject are $\tilde{f}(c_1) \leq \tilde{f}(c_3)$, $\tilde{f}(c_2) \leq \tilde{f}(c_3)$, and $\tilde{f}(c_4) \leq \tilde{f}(c_3)$ from the first round election, $\tilde{t}_{\{c_1,c_2,c_4\}}(c_1) \leq \tilde{t}_{\{c_1,c_2,c_4\}}(c_2)$ and $\tilde{t}_{\{c_1,c_2,c_4\}}(c_3) \leq \tilde{t}_{\{c_1,c_2,c_4\}}(c_2)$ from the second round election and $\tilde{t}_{\{c_1,c_4\}}(c_4) \leq \tilde{t}_{\{c_1,c_4\}}(c_1)$ from the final round. Assuming $\alpha = 0.05$ the ASNs for the first round are the same as calculated in Example 2. The ASNs for the remaining elections are 51.8, 64.0 and 1186 respectively. □

Example 4. The weakness of this naive approach is that inconsequential earlier elimination rounds can be difficult to audit even if they are irrelevant to the winner. Consider an election with five candidates c_1, c_2, c_3, c_4, c_5 and ballots (with multiplicity) $[c_1]$: 10000, $[c_2]$: 6000, $[c_3, c_2]$: 3000, $[c_3, c_1]$: 2000, $[c_4]$: 500, $[c_5]$: 499. The elimination order is $[c_5, c_4, c_3, c_2, c_1]$. Assuming $\alpha = 0.05$ then rejecting the null hypothesis that c_5 beat c_4 in the first round gives an ASN of $13,165,239$ indicating a full hand audit is required. But it is irrelevant to the election result. □

5.2 Simultaneous elimination

It is common in IRV elections to eliminate multiple candidates in a single round if it can be shown that the order of elimination cannot affect later rounds. Given an elimination order π we can simultaneously eliminate candidates $E = \{\pi(i)..\pi(i+k)\}$ if the sum of tallies of these candidates is less than the tally of the next lowest candidate. Let $C = \{\pi(i), \pi(i+1), \ldots \pi(k), \pi(k+1), \ldots \pi(n)\}$ be the set of candidates standing after the first $i-1$ have been eliminated. We can simultaneously eliminate E if:

$$t_C(c) > \sum_{c' \in E} t_C(c') \quad \forall c \in C \setminus E \tag{4}$$

This is because no matter which order the candidates in E are eliminated no candidate could ever garner a tally greater than one of the candidates in $C \setminus E$. Hence they will all be eliminated in any case. Note that since the remainder of the election only depends on the set of eliminated candidates and not their order, the simultaneous elimination can have no effect on later rounds of the election.

We can model the simultaneous elimination for auditing by considering all the simultaneously eliminated candidates E as as single loser l and rejecting hypotheses $\tilde{t}_C(c) \leq \tilde{t}_C(l)$ for each $c \in C \setminus E$. The statistic T_{wl} in this case is increased when we draw a ballot where w is the highest-ranked of remaining candidates C, and decreased when we draw a ballot where $c' \in E$ is the highest-ranked of remaining candidates C.

The elimination of all these null hypotheses is sufficient to prove that the multiple elimination is correct. This can then be combined with the audit of the rest of the elimination sequence, as described in Section 5.1, to test whether the election's announced winner is correct. Like the audit of a particular elimination sequence in Section 5.1, we are proving a stronger result than necessary, *i.e.* that a particular sequence of (possibly multiple) eliminations is valid, though there may be another way of getting the same candidate to win even if the multiple elimination isn't correct.

This often results in a much lower ASN, though not necessarily: sometimes the combined total of first preferences in E is very close to the next tally, so a lot of auditing is required. It may be better to audit each elimination individually in this case. It is possible to compute the ASN for each approach and choose the method that requires the least auditing, assuming the outcome is correct.

Example 5. Consider the election in Example 4. We can multiply eliminate the candidates $E = \{c_5, c_4\}$ since the sum of their tallies $499 + 500 < 5000$ which is the lowest tally of the other candidates. If we do this the difficult first round elimination auditing disappears. This shows the benefit of multiple elimination. The ASNs required for the joint elimination of E are 17.0, 36.2 and 49.1 as opposed to requiring a full hand audit.

Note that after this simultaneous elimination, the tallies for the three candidate election $\{c_1, c_2, c_3\}$ are c_1 : 10000, c_2 : 6000 and c_3 : 5000 and the ASNs to reject the hypotheses $\tilde{t}_{\mathcal{C}}(c_1) \leq \tilde{t}_{\mathcal{C}}(c_3)$ and $\tilde{t}_{\mathcal{C}}(c_2) \leq \tilde{t}_{\mathcal{C}}(c_3)$ are 77.6 and 1402 respectively.

Note we could also simultaneously eliminate the candidates $E = \{c_5, c_4, c_3\}$ since the sum of their tallies $499 + 500 + 5000 < 6000$ which is the lowest tally of the other candidate (that of c_2). But this will lead to a very difficult hypothesis to reject, $\tilde{t}_{\mathcal{C}}(c_2) \leq \tilde{t}_{\mathcal{C}}(\{c_5, c_4, c_3\})$ since the tallies are almost identical! The ASN is 158,156,493! This illustrates that multiple elimination may not always be beneficial. □

5.3 Winner only auditing

Up until now we consider auditing the entire IRV process to ensure that we are confident on all its outcomes. This is too strong since even if earlier eliminations happened in a different order it may not have any effect on the eventual winner.

Example 6. Consider an election with ballots $[c_1, c_2, c_3]$: 10000, $[c_2, c_1, c_3]$: 6000 and $[c_3, c_1, c_2]$: 5999. No simultaneous elimination is possible, and auditing that c_3 is eliminated before c_2 will certainly require a full hand audit. But even if c_2 were eliminated first it would not change the winner of the election. □

An alternate approach to ballot-polling RLAs for IRV elections is to simply reject the $n-1$ null hypotheses $\{\tilde{f}(w) \leq \tilde{t}_{\{w,l\}}(l)\}$ where w is the declared winner of the IRV election, and $l \in \mathcal{C} \setminus \{w\}$. This hypothesis states that l gets more votes than w where l is given the maximal possible votes it could ever achieve before w is eliminated, and w gets only its first round votes (the minimal possible votes it could ever hold). When we reject this hypothesis we are confident that there could not be any elimination order where w is eliminated before l. If all these hypotheses are rejected then we are assured that w is the winner of the election, independent of a particular elimination order.

Example 7. Consider the election of Example 6. We must reject the hypotheses that $\{\tilde{f}(c_1) \leq \tilde{t}_{\{c_1,c_2\}}(c_2)\}$ (c_1 is eliminated before c_2) and $\{\tilde{f}(c_1) \leq \tilde{t}_{\{c_1,c_2\}}(c_3)\}$ (c_1 is eliminated before c_3). The primary votes for c_1 are 10000, while the maximum votes that c_2 can achieve before c_1 is eliminated are 6000. Simultaneously the maximum votes that c_3 can achieve before c_1 is eliminated are 5999. Auditing to reject these hypotheses is not difficult. The ASNs are 98.4 and 98.3 ballots.

Note however that if the $[c_2, c_1, c_3]$ ballots were changed to be $[c_2, c_3, c_1]$ then the maximum votes that c_3 can achieve are 12000, and the hypothesis that (c_1 is eliminated before c_3) could not be rejected. Indeed in this case just changing a single vote could result in c_3 winning the election, so this election will need a full recount. □

There are, of course, some circumstances in which this does not work efficiently even though the margin of victory is large, for example if there are two runners-up who mostly (but not exclusively) preference each other.

Example 8. Consider an election with ballots $[c_1, c_2, c_3]$: 10000, $[c_2, c_3, c_1]$: 5000 $[c_2, c_1, c_3]$: 1500, $[c_3, c_2, c_1]$: 5000 and $[c_3, c_1, c_2]$: 500, and winner c_2. We cannot validate that c_2 won the election by a winner-only audit as we cannot reject the hypotheses that $\{\tilde{f}(c_2) \leq \tilde{t}_{\{c_2,c_1\}}(l)\}$. The winner's first preference tally is 6,500, while the total number of votes c_1 could have prior to c_2 being eliminated is 10,500. □

5.4 A general algorithm for finding efficient RLAs for IRV

This idea can be generalised to a method of choosing the set of facts that can be checked most efficiently (assuming no errors are found). We present an algorithm that achieves this by finding the easiest way to show that all election outcomes in which a candidate other than c_w won, did not arise, with a given level of statistical confidence.

Our algorithm, *audit-irv*, outlined in Figure 3, explores the tree of alternate elimination sequences, ending in a candidate $c' \neq c_w$. Each node is a partial (or complete) elimination sequence. For each node π, we consider the set of hypotheses that (i) can be proven with an application of BRAVO and (ii) any one of which disproves the outcome that π represents. We label each node π with the hypothesis h from this set that requires the least number of anticipated ballot polls (ASN) to prove, denoted $asn(h)$. We use the notation $h(\pi)$ and $asn(\pi)$ to represent the hypothesis assigned to π and the ASN for this hypothesis, respectively. Our algorithm finds a set of hypotheses to prove, denoted *audits*, that: validates the correctness of a given election outcome, with risk limit α; and for which the largest ASN of these hypothesis is minimised.

Note that our *risk-limit* follows directly from BRAVO: if the election outcome is wrong, then one of the facts in h must be false—a BRAVO audit with risk limit α will detect this with probability of at least $1 - \alpha$. However, our estimate of *efficiency* is only heuristic: ASNs for testing a single fact can be derived analytically, but the expected number of samples required to reject multiple hypothesis at once is very hard to compute, even if there are no discrepancies. We make a best guess based on the maximum ASN for any single fact—this is what we meant by "optimal" in this section, though it may not guarantee an optimally efficient audit overall. In Section 6 we describe simulated sample numbers for the results of our algorithm applied to real elections (assuming no discrepancies).

Consider a partial elimination sequence $\pi = [c, \ldots, w]$ of at least two candidates, leading to an alternate winner w. This sequence represents the suffix of a complete order – an outcome in which the candidates in $\mathcal{C} \setminus \pi$ have been previously eliminated, in some order. We define a function FindBestAudit(π, $\mathcal{C}$, $\mathcal{B}$, α) that finds the easiest to prove hypothesis (or fact) h, with the smallest ASN, which disproves the outcome π given risk limit α. For the outcome $\pi = [c|\ldots]$, FindBestAudit considers the following hypotheses:

WO(c,c'): Hypothesis that c beats $c' \in \pi$, for some $c' \in \pi, c' \neq c$, in a winner only audit of the form described in Section 5.3, with winner c and loser c', thus invalidating the sequence since c cannot be eliminated before c';

WO(c'',c): Hypothesis that $c'' \in \mathcal{C} \setminus \pi$ beats c in a winner only audit with winner c'' and loser c, thus invalidating the sequence since c'' cannot be eliminated before c;

IRV(c,c',$\{c'' \mid c'' \in \pi\}$): Hypothesis that c beats some $c' \neq c \in \pi$ in a BRAVO audit with winner c and loser c', under the assumption that the only candidates remaining are those in π (i.e. the set $\{c'' \mid c'' \in \pi\}$) with other candidates eliminated with their votes distributed to later preferences, thus invalidating the sequence since then c is not eliminated at this stage in an IRV election.

We assume that if no hypothesis exists with ASN less than $|\mathcal{B}|$ the function returns a dummy **INF** hypothesis with $ASN(\textbf{INF}) = +\infty$.

For an election with candidates $\mathcal{C}$ and winner c_w, *audit-irv* starts by adding $|\mathcal{C}| - 1$ partial elimination orders to an initially empty priority queue F, one for each alternate winner $c \neq c_w$ (Steps 4 to 9). The set *audits* is initially empty. For orders π containing a single candidate c, FindBestAudit considers the hypotheses **WO**(c'',c), candidate $c'' \neq c$ beats c in a winner only audit of the form described in Section 5.3, with winner c'' and loser c, for each $c'' \in \mathcal{C} \setminus \{c\}$. The hypothesis h with the smallest $ASN(h)$

```
audit-irv($\mathcal{C}$, $\mathcal{B}$, $c_w$, $\alpha$)
1  $audits \leftarrow \emptyset$
2  $F \leftarrow \emptyset$ ▷ $F$ is a set sequences to expand (the frontier)
3  $LB \leftarrow 0$
   ▷ Populate $F$ with single-candidate sequences
4  for each($c \in \mathcal{C} \setminus \{c_w\}$):
5      $\pi \leftarrow [c]$
6      $h \leftarrow$ FindBestAudit($\pi$, $\mathcal{C}$, $\mathcal{B}$, $\alpha$)
7      $hy[\pi] \leftarrow h$ ▷ Record best hypothesis for $\pi$
8      $ba[\pi] \leftarrow \pi$ ▷ Record best ancestor sequence for $\pi$
9      $F \leftarrow F \cup \{\pi\}$
   ▷ Repeatedly expand the sequence with largest ASN in $F$
10 while($|F| > 0$):
11     $\pi \leftarrow$ argmax$\{ASN(hy[\pi]) \mid \pi \in F\}$
12     $F \leftarrow F \setminus \{\pi\}$
13     if($ASN(hy[ba[\pi]]) \leq LB$):
14         $audits \leftarrow audits \cup \{hy[ba[\pi]]\}$
15         $F \leftarrow F \setminus \{\pi' \in F \mid ba[\pi]$ is a suffix of $\pi'\}$
16         continue
17     for each($c \in \mathcal{C} \setminus \pi$):
18         $\pi' \leftarrow [c] +\!\!+ \pi$
19         $h \leftarrow$ FindBestAudit($\pi'$, $\mathcal{C}$, $\mathcal{B}$, $\alpha$)
20         $hy[\pi'] \leftarrow h$
21         $ba[\pi'] \leftarrow$ if $ASN(h) < ASN(hy[ba[\pi]])$ then $\pi'$ else $ba[\pi]$
22         if($|\pi'| = |\mathcal{C}|$):
23             if($ASN(hy[ba[\pi']]) = \infty$):
24                 terminate algorithm, full recount necessary
25             else:
26                 $audits \leftarrow audits \cup \{hy[ba[\pi']]\}$
27                 $LB \leftarrow max(LB, ASN(hy[ba[\pi']]))$
28                 $F \leftarrow F \setminus \{\pi' \in F \mid ba[\pi]$ is a suffix of $\pi'\}$
29                 continue
30         else:
31             $F \leftarrow F \cup \{\pi'\}$
32 return $audits$ with maximum ASN equal to $LB$
```

Fig. 3: The *audit-irv* algorithm for searching for a collection of hypothesis to audit, with parallel applications of BRAVO, that validate the outcome of an IRV election with candidates $\mathcal{C}$, ballots $\mathcal{B}$, and winner c_w, with a given risk limit α.

is recorded in $hy[\pi]$. The best ancestor for π is recorded in $ba[\pi]$, for these singletons sequences it is always the sequence itself.

We repeatedly find and remove a partial sequence π in F for expansion (Steps 11 and 12). This is the sequence with the (equal) highest ASN. If the best ancestor for this sequence has an ASN lower than the current lower bound LB (Steps 13 to 16) we simply add the corresponding hypothesis to $audits$ and remove any sequences in F which are subsumed by this ancestor (have it as a suffix), and restart the main loop.

Otherwise (Steps 17 to 31) we create a new elimination sequence π' with c appended to the start of π ($[c] +\!\!+ \pi$) for each $c \in \mathcal{C} \setminus \pi$. For a new sequence π', FindBestAudit

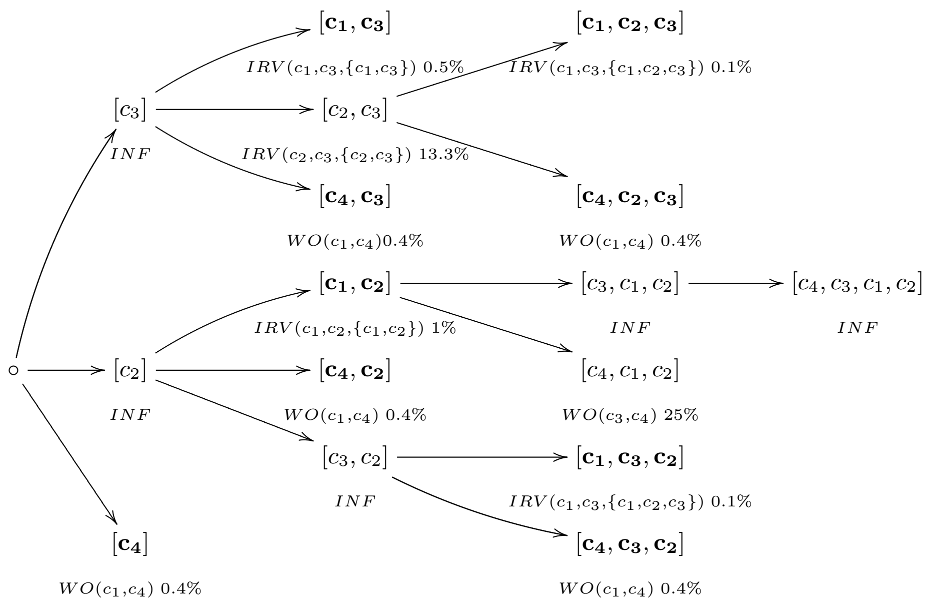

$[\mathbf{c_1},\mathbf{c_3}]$
$IRV(c_1,c_3,\{c_1,c_3\})$ 0.5%
$[\mathbf{c_1},\mathbf{c_2},\mathbf{c_3}]$
$IRV(c_1,c_3,\{c_1,c_2,c_3\})$ 0.1%
$[c_3]$
INF
$[c_2,c_3]$
$IRV(c_2,c_3,\{c_2,c_3\})$ 13.3%
$[\mathbf{c_4},\mathbf{c_3}]$
$WO(c_1,c_4)$0.4%
$[\mathbf{c_4},\mathbf{c_2},\mathbf{c_3}]$
$WO(c_1,c_4)$ 0.4%
$[\mathbf{c_1},\mathbf{c_2}]$
$IRV(c_1,c_2,\{c_1,c_2\})$ 1%
$[c_3,c_1,c_2]$
INF
$[c_4,c_3,c_1,c_2]$
INF
○
$[c_2]$
INF
$[\mathbf{c_4},\mathbf{c_2}]$
$WO(c_1,c_4)$ 0.4%
$[c_4,c_1,c_2]$
$WO(c_3,c_4)$ 25%
$[c_3,c_2]$
INF
$[\mathbf{c_1},\mathbf{c_3},\mathbf{c_2}]$
$IRV(c_1,c_3,\{c_1,c_2,c_3\})$ 0.1%
$[\mathbf{c_4},\mathbf{c_3},\mathbf{c_2}]$
$WO(c_1,c_4)$ 0.4%
$[\mathbf{c_4}]$
$WO(c_1,c_4)$ 0.4%

Fig. 4: Tree formed by *audit-irv* for the election of Example 9. The best hypothesis for each sequence is shown below the sequence, together with the ASN. The selected frontier is shown in bold which requires the audits: $IRV(c_1, c_3, \{c_1, c_3\})$, $IRV(c_1, c_3, \{c_1, c_2, c_3\})$, $WO(c_1, c_4)$, $IRV(c_1, c_2, \{c_1, c_2\})$.

finds the hypothesis h requiring the least auditing effort to prove. We record (Step 20) this as the hypothesis for $hy[\pi'] = h$. We calculate (Step 21) the best ancestor of π' by comparing the ASN for its hypothesis with that of its ancestor.

If the sequence π' is complete, then we known one of its ancestors (including itself) must be audited. If the best of these is infinite, we terminate, a full recount is necessary. Otherwise we add the hypothesis of its best ancestor to $audits$ and remove all sequences in F which are subsumed by this ancestor. If the sequence is not complete we simply add it into the set of sequences to be expanded F.

Example 9. Consider an election with ballots $[c_1, c_2, c_3]$: 5000, $[c_1, c_3, c_2]$: 5000 $[c_2, c_3, c_1]$: 5000, $[c_2, c_1, c_3]$: 1500, $[c_3, c_2, c_1]$: 5000, $[c_3, c_1, c_2]$: 500, and $[c_4, c_1]$: 5000, and candidates c_1 to c_4. The initial tallies are: c_1: 10000; c_2: 6500; c_3: 5500; c_4: 5000. Candidates c_4, c_3, and c_2 are eliminated, in that order, with winner c_1. In a winner only audit ($\alpha = 0.05$), we cannot show that c_1 beats c_3, or that c_1 beats c_2, as c_1's first preference tally (of 10000 votes) is less than the total number of ballots that we could attribute to c_2 and c_3 (11500 and 10500, respectively). Simultaneous elimination is not applicable in this instance, as no sequences of candidates can be eliminated in a group. In an audit of the whole elimination order (as per Section 5.1), the loss of c_4 to c_1, c_2, and c_3 is the most challenging to audit. The ASN for this audit is 25% of all ballots.

Our *audit-irv* algorithm, however, finds a set of hypotheses that can be proven with a maximum ASN of 1% (with $\alpha = 0.05$), and that consequently rule out all elimination sequences that end in a candidate other than c_1. This audit proves the hypotheses: c_1 beats c_2 if c_3 and c_4 have been eliminated (ASN of 1%); c_1 beats c_3 if c_2 and c_4 have been eliminated (ASN 0.5%); c_1 beats c_4 in a winner only audit (ASN 0.4%); and that c_1 beats c_3 if c_4 has been eliminated (ASN 0.1%). Figure 4 shows the final state of the tree explored by *audit-irv*. We record, under each sequence, the easiest hypothesis that, if proven, *disproves* an outcome ending in that sequence (alongside its ASN). The hypotheses underneath each leaf node (excluding duplicates) form our audit. Once *audit-irv* creates the node $[c_4, c_3, c_1, c_2]$ and finds that it cannot disprove this hypothesis, all descendants of $[c_1, c_2]$ are pruned from the tree. At this stage, LB is equal to 1%, and all leaves can be disproved with an ASN $\leq LB$ and the algorithm terminates. □

6 Computational Results

We have simulated the audits described in Section 5.1 (auditing the elimination order, EO), Section 5.2 (auditing with simultaneous elimination, SE), and Section 5.3 (winner only auditing, WO), on 21 US IRV elections held between 2007 and 2014, and on the IRV elections held across 93 electorates in the 2015 state election in New South Wales (NSW), Australia. We report the average number of ballot polls (expressed as a percentage of ballots cast) required to complete each of these audits, for varying risk limits, in Table 2, alongside the ASN of each audit. Each audit is run 10 times, using 10 different random seeds to control the sequence of ballots polled, and the number of ballots polled averaged over those runs. For brevity, we include the results for only a portion of the NSW seats, with the full results provided in the full version of this paper. The margin of victory (MOV) for each election is computed using the algorithm of [3].

All experiments have been conducted on a machine with an Intel Xeon Platinum 8176 chip (2.1GHz), and 1TB of RAM.

Table 2 shows that performing a winner only audit can be much easier than auditing the full elimination order (with or without the use of simultaneous elimination). This is the case for the 2013 Minneapolis Mayor, 2014 Oakland Mayor, and the 2010 Oakland D4 City Council elections. In some cases, winner only audits are more challenging (or not possible) as we seek to show that a candidate c (on just their first preference votes) could have beaten another c' (who is given all votes in which they appear before c or in which they appear, but c does not). Even if c does beat c' in the true outcome of the election, this audit may not be able to prove this (see Pierce 2008 County Executive, Oakland 2012 D5 City Council, and Aspen 2009 Mayor for examples). Auditing with simultaneous elimination (grouping several eliminated candidates into a single 'super' candidate) can be more efficient than auditing each individual elimination (see Berkeley 2010 D8 City Council, Berkeley 2012 Mayor, Oakland 2010 Mayor, San Francisco 2007 Mayor, and Sydney NSW). In some instances, however, the tally of the super candidate is quite close to that of the next eliminated candidate, resulting in a challenging audit (see Campbelltown NSW, and Berkeley 2010 D4 City Council).

Table 3 reports the maximum ASN of the audit found by *audit-irv* for each of the 26 elections examined in Table 2, alongside the ASN and average actual ballot polls

				EO				**SE**				**WO**			
				α 0.01		α 0.05		α 0.01		α 0.05		α 0.01		α 0.05	
Election	$\lvert\mathcal{C}\rvert$	$\lvert\mathcal{B}\rvert$	**MOV**	Polls %	ASN %	Polls %	ASN %	Polls %	ASN %	Polls %	ASN %	Polls %	ASN %	Polls %	ASN %
Berkeley 2010 D7 CC	4	4,682	364 (7%)	6.7	7.2	3.9	4.7	7.5	7.2	4	4.7	8.7	22.4	4.9	14.7
Berkeley 2010 D8 CC	4	5,333	878 (16%)	∞	∞	∞	∞	2.9	4.2	2	2.8	1.3	1.8	0.8	1.2
Oakland 2010 D6 CC	4	14,040	2,603 (19%)	4.0	4.4	3	2.9	0.7	0.9	0.5	0.6	0.4	0.5	0.3	0.3
Pierce 2008 CC	4	43,661	2,007 (5%)	3.1	2.2	1.8	1.4	3.1	2.2	1.8	1.4	3.2	4.1	1.8	2.7
Pierce 2008 CAD	4	159,987	8,396 (5%)	0.3	0.5	0.2	0.3	0.3	0.5	0.2	0.3	0.5	1.2	0.3	0.8
Aspen 2009 Mayor	5	2,544	89 (4%)	62.4	71.8	52.7	46.9	62.4	71.8	54.8	46.9	∞	∞	∞	∞
Berkeley 2010 D1 CC	5	6,426	1,174 (18%)	2.4	1.7	1.6	1.1	2.4	1.7	1.6	1.1	1.1	1.1	0.8	0.7
Berkeley 2010 D4 CC	5	5,708	517 (9%)	7.5	7	6	4.7	28.7	40.7	17.8	26.6	4.9	7.3	3.8	4.8
Oakland 2012 D5 CC	5	13,482	486 (4%)	11.2	10.3	7.3	6.7	15.1	10.3	11.8	6.7	∞	∞	∞	∞
Pierce 2008 CE	5	312,771	2,027 (1%)	11.6	15.1	7.6	9.8	11.6	15.1	7.6	9.8	∞	∞	∞	∞
San Leandro 2012 D4 CC	5	28,703	2,332 (8%)	9.3	9.7	6.3	6.3	9.3	9.7	6.3	6.3	1.1	4.4	0.8	2.9
Oakland 2012 D3 CC	7	26,761	386 (1%)	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
Pierce 2008 CAS	7	312,771	1,111 (0.4%)	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
San Leandro 2010 Mayor	7	23,494	116 (0.5%)	∞	∞	92.9	∞	∞	∞	92.9	∞	∞	∞	∞	∞
Berkeley 2012 Mayor	8	57,492	8,522 (15%)	94.6	∞	77	∞	2.3	2.6	1.6	1.7	0.2	0.2	0.1	0.2
Oakland 2010 D4 CC	8	23,884	2,329 (10%)	∞	∞	76.4	∞	∞	∞	∞	∞	0.9	3.1	0.6	2
Aspen 2009 CC	11	2,544	35 (1%)	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
Oakland 2010 Mayor	11	122,268	1,013 (1%)	∞	∞	∞	∞	21.5	23.8	15	15.5	∞	∞	∞	∞
Oakland 2014 Mayor	11	101,431	10,201 (10%)	∞	∞	∞	∞	∞	∞	∞	∞	0.8	19.8	0.5	12.9
San Francisco 2007 Mayor	18	149,465	50,837 (34%)	∞	∞	∞	∞	0.03	0.03	0.02	0.02	0.01	0.01	0.01	0.01
Minneapolis 2013 Mayor	36	79,415	6,949 (9%)	∞	∞	∞	∞	∞	∞	∞	∞	0.5	3.1	0.3	2.1
Balmain NSW 2015	7	46,952	1,731 (3.7%)	∞	∞	∞	∞	83.8	∞	65.4	82	5.2	31.6	3.7	20.6
Campbelltown NSW 2015	5	45,124	3,096 (6.9%)	13.6	12.2	8.4	8	∞	∞	∞	∞	1.3	1.7	0.9	1.1
Gosford NSW 2015	6	48,259	102 (0.2%)	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
Lake Macquarie NSW 2015	7	47,698	4,253 (8.9%)	27.7	22.8	14.5	15	6.9	7.8	3.2	5.1	0.7	1.6	0.5	1
Sydney NSW 2015	8	42,747	2,864 (6.7%)	∞	∞	∞	∞	3.3	4.6	2.2	3	1.6	6.9	1	4.5

Table 2: Average ballot polls performed (as a percentage of ballots cast) over 10 simulated audits of 26 IRV elections using a series of different auditing methods (with an α of 0.01 and 0.05): auditing the elimination order (EO); auditing with simultaneous elimination (SE); and winner only auditing (WO). Also reported is each elections margin of victory (MOV). The notation ∞ indicates a percentage of ballots (or ASN) greater than 100%. CC, CE, CAD, and CAS denote City Council, County Executive, County Auditor, and County Assessor.

required across 10 simulations of the best alternative audit (elimination order EO, simultaneous elimination SE, and winner only WO). Also reported is the runtime (in seconds) of *audit-irv* and the number nodes expanded (note that this does not include the creation of nodes forming the initial queue). Our *audit-irv* algorithm finds an audit (a collection of facts to prove by simultaneous applications of BRAVO) with an ASN that is equal to or lower than the ASN of the best alternative. The Oakland 2012 D3 City Council and Pierce 2008 County Assessor elections are particularly interesting. We are able to find a method of auditing the outcome of these elections that is significantly easier than the EO, SE, and WO methods, which suggest a full recount. The ASN is just an estimate, however, and the actual auditing effort required may deviate from this. For the Balmain NSW election, for example, the ASN of the best alternative audit (WO) is 20.6%. The average actual number of ballot polls required is 3.7% of the total, across 10 simulations of the audit. The ASN and actual audit effort required for the *audit-irv* audit in this instance is 1.9% and 3.2%, respectively. For the Oakland 2014 Mayor election, the ASN of the best alternative audit (WO) is 12.9% while the average actual auditing effort required is 0.5%. In contrast, the ASN of the audit found by *audit-irv* is 0.1% while the average actual effort required is 5.4%.

7 Conclusion

This paper provides a comprehensive, practical method of conducting risk-limiting ballot-polling audits for IRV. We use Stark's BRAVO as a black box, and show how to combine facts together to audit an IRV outcome. Most can be audited very efficiently. This algorithm dominates other approaches to auditing IRV elections. Over a collection of parliamentary seats or council races, most outcomes could be confirmed quickly with very little effort, while others would require some more careful auditing, and those with very small margins could be identified immediately and sent for a full manual recount.

References

1. T. Antonyan, S. Davtyan, S. Kentros, A. Kiayias, L. Michel, N. Nicolaou, A. Russell, and A. A. Shvartsman. State-wide elections, optical scan voting systems, and the pursuit of integrity. *IEEE Transactions on Information Forensics and Security*, 4(4):597–610, 2009.
2. B. Beckert, M. Kirsten, V. Klebanov, and C. Schürmann. Automatic margin computation for risk-limiting audits. In *International Joint Conference on Electronic Voting*, pages 18–35. Springer, 2016.
3. M. Blom, P. J. Stuckey, V. Teague, and R. Tidhar. Efficient Computation of Exact IRV Margins. In *European Conference on AI (ECAI)*, pages 480–487, 2016.
4. J.L. Hall, L.W. Miratrix, P.B. Stark, M. Briones, E. Ginnold, F. Oakley, M. Peaden, G. Pellerin, T. Stanionis, and T. Webber. Implementing risk-limiting post-election audits in California. In *Proc. 2009 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE '09)*, Montreal, Canada, August 2009. USENIX.
5. J. A. Kroll, J. A. Halderman, and E. W. Felten. Efficiently auditing multi-level elections. *Ann Arbor*, 1001:48109.
6. M. Lindeman and P.B. Stark. A gentle introduction to risk-limiting audits. *IEEE Security and Privacy*, 10:42–49, 2012.

	Best Alt.			*audit-irv* ($\alpha = 0.05$)			
Election	**Audit**	**Polls %**	**ASN %**	**Polls %**	**ASN %**	**Time (s)**	**Exp.**
Berkeley 2010 D7 CC	EO	3.9	4.7	5.4	4.7	0.003	3
Berkeley 2010 D8 CC	WO	0.8	1.2	0.9	0.9	0.01	6
Oakland 2010 D6 CC	WO	0.3	0.3	0.3	0.3	0.01	3
Pierce 2008 CC	EO,SE	1.8	1.4	1.5	1.4	0.03	3
Pierce 2008 CAD	EO,SE	0.2	0.3	0.3	0.3	0.1	3
Aspen 2009 Mayor	EO	52.7	46.9	28.1	46.9	0.01	9
Berkeley 2010 D1 CC	WO	0.8	0.7	0.6	0.6	0.01	5
Berkeley 2010 D4 CC	WO	3.8	4.8	1.6	2.7	0.01	5
Oakland 2012 D5 CC	EO	7.3	6.7	5.2	6.7	0.02	5
Pierce 2008 CE	EO,SE	7.6	9.8	13.9	9.8	0.9	10
San Leandro 2012 D4 CC	WO	0.8	2.9	0.8	0.6	0.06	8
Oakland 2012 D3 CC	**–**	∞	∞	**14.2**	**13.1**	**0.2**	**20**
Pierce 2008 CAS	**–**	∞	∞	**17**	**22.7**	**3.4**	**28**
San Leandro 2010 Mayor	EO,SE	92.9	∞	87.6	∞	0.08	8
Berkeley 2012 Mayor	WO	0.1	0.2	0.1	0.1	0.3	14
Oakland 2010 D4 CC	WO	0.6	2	0.6	0.5	0.3	15
Aspen 2009 CC	–	∞	∞	∞	∞	0.4	172
Oakland 2010 Mayor	SE	15	15.5	15.3	15.5	2.7	44
Oakland 2014 Mayor	WO	0.5	12.9	5.4	0.1	106	606
San Francisco 2007 Mayor	WO	0.01	0.01	0.01	0.01	23	130
Minneapolis 2013 Mayor	WO	0.3	2.1	0.2	0.2	10.8	43
Balmain NSW 2015	WO	3.7	20.6	3.2	1.9	0.2	8
Campbelltown NSW 2015	WO	0.9	1.1	0.8	0.7	0.1	5
Gosford NSW 2015	–	∞	∞	∞	∞	0.1	6
Lake Macquarie NSW 2015	WO	0.5	1	0.5	0.3	0.2	8
Sydney NSW 2015	SE	1	4.5	1.3	0.7	0.2	11

Table 3: ASN, and average ballot polls required across 10 simulations, of audit found by *audit-irv* for 26 IRV elections, alongside best known alternative (EO, SE, or WO) given a risk limit α of 0.05. Notation ∞ indicates a percentage of ballots (or ASN) greater than 100%. 'Exp' denotes the number of node expansions performed by *audit-irv*.

7. M. Lindeman, P.B. Stark, and V. Yates. BRAVO: Ballot-polling risk-limiting audits to verify outcomes. In *Proceedings of the 2011 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections (EVT/WOTE '11)*. USENIX, 2012.
8. T.R. Magrino, R.L. Rivest, E. Shen, and D.A. Wagner. Computing the margin of victory in IRV elections. In *USENIX Accurate Electronic Voting Technology Workshop: Workshop on Trustworthy Elections*, USENIX Association Berkeley, CA, USA, 2011.
9. A.D. Sarwate, S. Checkoway, and H. Shacham. Risk-limiting audits and the margin of victory in nonplurality elections. *Politics, and Policy*, 3(3):29–64, 2013.
10. P. B. Stark. A sharper discrepancy measure for post-election audits. *The Annals of Applied Statistics*, pages 982–985, 2008.

Risk-Limiting Audits by Stratified Union-Intersection Tests of Elections (SUITE)

Kellie Ottoboni[1] [0000−0002−9107−3402], Philip B. Stark[1] [0000−0002−3771−9604], Mark Lindeman[2] [0000−0001−8815−815X], and Neal McBurnett [0000−0001−8667−1830]

[1] Department of Statistics, University of California, Berkeley, CA, USA
[2] Verified Voting Foundation

Abstract. Risk-limiting audits (RLAs) offer a statistical guarantee: if a full manual tally of the paper ballots would show that the reported election outcome is wrong, an RLA has a known minimum chance of leading to a full manual tally. RLAs generally rely on random samples. Stratified sampling—partitioning the population of ballots into disjoint strata and sampling independently from the strata—may simplify logistics or increase efficiency compared to simpler sampling designs, but makes risk calculations harder. We present SUITE, a new method for conducting RLAs using stratified samples. SUITE considers all possible partitions of outcome-changing error across strata. For each partition, it combines P-values from stratum-level tests into a combined P-value; there is no restriction on the tests used in different strata. SUITE maximizes the combined P-value over all partitions of outcome-changing error. The audit can stop if that maximum is less than the risk limit. Voting systems in some Colorado counties (comprising 98.2% of voters) allow auditors to check how the system interpreted each ballot, which allows *ballot-level comparison* RLAs. Other counties use *ballot polling*, which is less efficient. Extant approaches to conducting an RLA of a statewide contest would require major changes to Colorado's procedures and software, or would sacrifice the efficiency of ballot-level comparison. SUITE does not. It divides ballots into two strata: those cast in counties that can conduct ballot-level comparisons, and the rest. Stratum-level P-values are found by methods derived here. The resulting audit is substantially more efficient than statewide ballot polling. SUITE is useful in any state with a mix of voting systems or that uses stratified sampling for other reasons. We provide an open-source reference implementation and exemplar calculations in Jupyter notebooks.

Keywords: stratified sampling, nonparametric tests, Fisher's combining function, sequential hypothesis tests, Colorado risk-limiting audits, maximizing P-values over nuisance parameters, union-intersection test, intersection-union test

Acknowledgements. We are grateful to Ronald L. Rivest and Steven N. Evans for helpful conversations and suggestions.

1 Introduction

A risk-limiting audit (RLA) of an election contest is a procedure that has a known minimum chance of leading to a full manual tally of the ballots if the electoral outcome according to that tally would differ from the reported outcome. *Outcome* means the winner(s) (or, for instance, whether there is a runoff)—not the numerical vote totals. RLAs require a durable, voter-verifiable record of voter intent, such as paper ballots, and they assume that this audit trail is sufficiently complete and accurate that a full hand tally would show the true electoral outcome. That assumption is not automatically satisfied: a *compliance audit* [16] is required to check whether the paper trail is trustworthy.

Current methods for risk-limiting audits are generally *sequential hypothesis testing procedures*: they examine more ballots, or batches of ballots, until either (i) there is strong statistical evidence that a full hand tabulation would confirm the outcome, or (ii) the audit has led to a full hand tabulation, the result of which should become the official result.

RLAs have been conducted in California, Colorado, Ohio, and Denmark, and are required by law in Colorado (CRS 1-7-515) and Rhode Island (SB 413A and HB 5704A).

The most efficient and transparent sampling design for risk-limiting audits selects individual ballots uniformly at random, with or without replacement [13]. Risk calculations for such samples can be made simple without sacrificing rigor [14,6]. However, to audit contests that cross jurisdictional boundaries then requires coordinating sampling in different counties, and may require different counties to use the lowest-common denominator method for assessing risk from the sample, which would not take full advantage of the capabilities of some voting systems. For instance, any system that uses paper ballots as the official record can conduct *ballot-polling* audits, while *ballot-level comparison audits* require systems to generate *cast-vote records* that can be checked manually against a human reading of the paper [5,6]. (These terms are described in Section 3.)

Stratified RLAs have been considered previously, primarily to conform with legacy audit laws under which counties draw audit samples independently of each other, but also to allow auditors to start the audit before all vote-by-mail or provisional ballots have been tallied, by sampling independently from ballots cast in person, by mail, and provisionally, as soon as subtotals for each group are available [9,4]. However, extant methods address only a single approach to auditing, batch-level comparisons, and only a particular test statistic.

Here, we introduce SUITE, a more general approach to conducting RLAs using stratified samples. SUITE is a twist on *intersection-union* tests [7], which represent the null hypothesis as the intersection of a number of simpler hypotheses, and the alternative hypothesis as a union of their alternatives. In contrast, here, the null is the union of simpler hypotheses, and the alternative is the intersection of their alternatives. The approach involves finding the maximum P-value over a vector of nuisance parameters that describe the simple hypotheses, all allocations of tabulation error across strata for which a full count would find a different electoral outcome than was reported. (A *nuisance parameter* is a property of the population that is not of direct interest, but that affects the probability distribution of the data. *Overstatement* is error that made the margin of one or more winners over one or more losers appear larger than it really was. The total *overstatement* across strata determines whether the reported outcome is correct; the overstatements in individual strata are nuisance parameters that affect the distribution of the audit sample.)

The basic building block for the method is testing whether the overstatement error in a single stratum exceeds a quota. Fisher's combining function is used to merge P-values for tests in different strata into a single P-value for the hypothesis that the overstatement in every stratum exceeds its quota. If that hypothesis can be rejected for *all* stratum-level quotas that could change the outcome—that is, if the maximum combined P-value is sufficiently small—the audit can stop.

It is not actually necessary to consider all possible quotas: the P-value involves a sum of monotonic functions, which allows us to find upper and lower bounds everywhere using only values on a discrete grid. We present a numerical procedure, implemented in Python, to find bounds on the maximum P-value when there are two strata. The procedure can be generalized to more than two strata.

Section 2 presents the new approach to stratified auditing. Section 3 illustrates the method by solving a problem pertinent to Colorado: combining ballot-polling in one stratum with ballot-level comparisons in another. This requires straightforward modifications to the mathematics behind ballot-polling and ballot-level comparison to allow the overstatement to be compared to specified thresholds other than the overall contest margin; those modifications are described in Sections 3.1 and 3.2. Section 4 gives numerical examples of simulated audits, using parameters intended to reflect how the procedure would work in Colorado. We provide example software implementing the risk calculations for our recommended approach in Python Jupyter notebooks.[3] Section 5 gives recommendations and considerations for implementation.

2 Stratified audits

Stratified sampling involves partitioning a population into non-overlapping groups and drawing independent random samples from those groups. [9,4] developed RLAs based on comparing stratified samples of batches of ballots to hand counts of the votes in those batches: batch-level comparison RLAs, using a particular test statistic. The method we develop here is more general and more flexible: it can be used with any test statistic, and test statistics in different strata need not be the same—which is key to combining audits of ballots cast using diverse voting technologies.

Here and below, we consider auditing a single plurality contest at a time, although the same sample can be used to audit more than one contest (and super-majority contests), and there are ways of combining audits of different

[3] See https://github.com/pbstark/CORLA18.

contests into a single process [10,14]. We use terminology drawn from a number of papers, notably [6].

An *overstatement error* is an error that caused the margin between *any* reported winner and *any* reported loser to appear larger than it really was. An *understatement error* is an error that caused the margin between *every* reported winner and *every* reported loser to appear to be smaller than it really was. Overstatements cast doubt on outcomes; understatements do not, even though they are tabulation errors.

We use w to denote a reported winner and ℓ to denote a reported loser. The total number of reported votes for candidate w is V_w and the total for candidate ℓ is V_ℓ. Thus $V_w > V_\ell$, since w is reported to have gotten more votes than ℓ.

Let $V_{w\ell} \equiv V_w - V_\ell > 0$ denote the contest-wide margin (in votes) of w over ℓ. We have S strata. Let $V_{w\ell,s}$ denote the margin (in votes) of reported winner w over reported loser ℓ in stratum s. Note that $V_{w\ell,s}$ might be negative in one stratum, but $\sum_{s=1}^{S} V_{w\ell,s} = V_{w\ell} > 0$. Let $A_{w\ell}$ denote the margin (in votes) of reported winner w over reported loser ℓ that a full hand count would show: the *actual* margin, in contrast to the *reported* margin $V_{w\ell}$. Reported winner w really beat reported loser ℓ if and only if $A_{w\ell} > 0$. Define $A_{w\ell,s}$ to be the actual margin (in votes) of w over ℓ in stratum s.

Let $\omega_{w\ell,s} \equiv V_{w\ell,s} - A_{w\ell,s}$ be the *overstatement* of the margin of w over ℓ in stratum s. Reported winner w really beat reported loser ℓ if and only if $\omega_{w\ell} \equiv \sum_s \omega_{w\ell,s} < V_{w\ell}$.

An RLA is a test of the hypothesis that the outcome is wrong, that is, that w did not really beat ℓ: $\sum_s \omega_{w\ell,s} \geq V_{w\ell}$. The null is true if and only if there exists *some* S-tuple of real numbers $(\lambda_s)_{s=1}^{S}$ with $\sum_s \lambda_s = 1$ such that $\omega_{w\ell,s} \geq \lambda_s V_{w\ell}$ for all s.[4] Thus if we can reject the conjunction hypothesis $\cap_s\{\omega_{w\ell,s} \geq \lambda_s V_{w\ell}\}$ at significance level α for all (λ_s) such that $\sum_s \lambda_s = 1$, we can stop the audit, and the risk limit will be α.

2.1 Fisher's combination method

Fix $\lambda \equiv (\lambda_s)_{s=1}^{S}$, with $\sum_s \lambda_s = 1$. To test the conjunction hypothesis that stratum null hypotheses are true, that is, that $\omega_{w\ell,s} \geq \lambda_s V_{w\ell}$ for all s, we use Fisher's combining function. Let $p_s(\lambda_s)$ be the P-value of the hypothesis $\omega_{w\ell,s} \geq \lambda_s V_{w\ell}$. If the null hypothesis is true, then

$$\chi(\lambda) = -2 \sum_{s=1}^{S} \ln p_s(\lambda_s) \tag{1}$$

has a probability distribution that is dominated by the chi-square distribution with $2S$ degrees of freedom.[5] Fisher's combined statistic will tend to be small

[4] "If" is straightforward. For "only if," suppose $\omega_{w\ell} \geq V_{w\ell}$. Set $\lambda_s = \frac{\omega_{w\ell,s}}{\sum_t \omega_{w\ell,t}}$. Then $\sum_s \lambda_s = 1$, and $\omega_{w\ell,s} = \lambda_s \omega_{w\ell} \geq \lambda_s V_{w\ell}$ for all s.

[5] If the stratum-level tests had continuously distributed P-values, the distribution would be exactly chi-square with $2S$ degrees of freedom, but if any of the P-values

when all stratum-level null hypotheses are true. If any is false, then as the sample size increases, Fisher's combined statistic will tend to grow.

If, for all λ with $\sum_s \lambda_s = 1$, we can reject the conjunction hypothesis at level α, (i.e., if the minimum value of Fisher's combined statistic over all λ is larger than the $1-\alpha$ quantile of the chi-square distribution with $2S$ degrees of freedom), the audit can stop.

If the audit is allowed to "escalate" in steps, increasing the sample size sequentially, then either the tests used in the separate strata have to be sequential tests, or multiplicity needs to be taken into account, for instance by adjusting the risk limit at each step. Otherwise, the overall procedure can have a risk limit that is much larger than α. For examples of controlling for multiplicity when using non-sequential testing procedures in an RLA, see [9,11].

The stratum-level P-value $p_s(\lambda)$ could be a P-value for the hypothesis $\omega_{w\ell,s} \geq \lambda_s V_{w\ell}$ from any test procedure. We assume, however, that p_s is based on a one-sided test, and that the tests for different values of λ "nest" in the sense that if $a > b$, then $p_s(a) > p_s(b)$. This monotonicity is a reasonable requirement because the evidence that the overstatement is greater than a should be weaker than the evidence that the overstatement is greater than b, if $a > b$. In particular, this monotonicity holds for the tests proposed in Sections 3.1 and 3.2.

One could use a function other than Fisher's to combine the stratum-level P-values into a P-value for the conjunction hypothesis, provided it satisfies these properties (see [7]):

- the function is non-increasing in each argument and symmetric with respect to rearrangements of the arguments
- the combining function attains its supremum when one of the arguments approaches zero
- for every level α, the critical value of the combining function is finite and strictly smaller than the function's supremum.

For instance, one could use Liptak's function, $T = \sum_i \Phi^{-1}(1-p_i)$, or Tippett's function, $T = \max_i(1-p_i)$.

Fisher's function is convenient for this application because the tests in different strata are independent, so the chi-squared distribution dominates the distribution of $\chi(\cdot)$ when the null hypothesis is true. If tests in different strata were correlated, the null distribution of the combination function would need to be calibrated by simulation; some other combining function might have better properties than Fisher's [7].

2.2 Maximizing Fisher's combined P-value for $S = 2$

We now specialize to $S = 2$ strata. The set of $\lambda = (\lambda_1, \lambda_2)$ such that $\sum_s \lambda_s = 1$ is then a one-dimensional family: if $\lambda_1 = \lambda$, then $\lambda_2 = 1-\lambda$. For a given set of data,

has atoms when the null hypothesis is true, it is in general stochastically smaller. This follows from a coupling argument along the lines of Theorem 4.12.3 in [3].

finding the maximum P-value over all λ is thus a one-dimensional optimization problem. We provide two software solutions to the problem.

The first approach approximates the maximum via a grid search, refining the grid once the maximum has been bracketed. This is not guaranteed to find the global maximum exactly, although it can approximate the maximum as closely as one desires by refining the mesh, since the objective function is continuous.

The second, more rigorous approach uses bounds on Fisher's combining function χ for all λ. (A lower bound on χ implies an upper bound on the P-value: if, for all λ, the lower bound is larger than the $1-\alpha$ quantile of the chi-squared distribution with 4 degrees of freedom, the maximum P-value is no larger than α.)

Some values of λ can be ruled out *a priori*, because (for instance) $\omega_{w\ell,s} \leq V_{w\ell,s} + N_s$, where N_s is the number of ballots cast in stratum s, and thus

$$1 - \frac{V_{w\ell,2} + N_2}{V_{w\ell}} \leq \lambda \leq \frac{V_{w\ell,1} + N_1}{V_{w\ell}}. \tag{2}$$

Let λ_- and λ_+ be lower and upper bounds on λ.

Recall that $p_s(\cdot)$ are monotonically increasing functions, so, as a function of λ, $p_1(\lambda)$ increases monotonically and $p_2(1-\lambda)$ decreases monotonically. Suppose $[a, b) \subset [\lambda_-, \lambda_+]$. Then for all $\lambda \in [a, b)$, $-2 \ln p_1(\lambda) \geq -2 \ln p_1(b)$ and $-2 \ln p_2(1-\lambda) \geq -2 \ln p_2(1-a)$. Thus

$$\chi(\lambda) = -2(\ln p_1(\lambda) + \ln p_2(1-\lambda)) \geq -2(\ln p_1(b) + \ln p_2(1-a)) \equiv \chi_-[a, b). \tag{3}$$

This gives a lower bound for χ on the interval $[a, b)$; the corresponding upper bound is $\chi(\lambda) \leq -2(\ln p_1(a) + \ln p_2(1-b)) \equiv \chi_+[a, b)$. Partitioning $[\lambda_-, \lambda_+]$ into a collection of intervals $[a_k, a_{k+1})$ and finding $\chi_-[a_k, a_{k+1})$ and $\chi_+[a_k, a_{k+1})$ for each yields piecewise-constant lower and upper bounds for $\chi(\lambda)$.

If, for all $\lambda \in [\lambda_-, \lambda_+]$, the lower bound on χ is larger than the $1-\alpha$ quantile of the chi-square distribution with 4 degrees of freedom, the audit can stop. On the other hand, if for some $\lambda \in [\lambda_-, \lambda_+]$, the upper bound is less than the $1-\alpha$ quantile of the chi-square distribution with 4 degrees of freedom, or if $\chi(a_k)$ is less than this quantile at any grid point $\{a_k\}$, the sample size in one or both strata needs to increase. If the lower bound is less than the $1-\alpha$ quantile on some interval, but $\chi(a_k)$ is above this quantile at every grid point $\{a_k\}$, then one should improve the lower bound by refining the grid and/or by increasing the sample size in one or both strata.

3 Auditing cross-jurisdictional contests

As mentioned above, stratified sampling can simplify audit logistics by allowing jurisdictions to sample ballots independently of each other, or by allowing a single jurisdiction to sample independently from different collections of ballots (e.g., vote-by-mail versus cast in person). SUITE allows stratified samples to be combined into an RLA of contests that include ballots from more than one stratum.

We present an example where SUITE is helpful for a different reason: it enables an RLA to take advantage of differences among voting systems to reduce audit sample sizes, which solves a current problem in Colorado.

CRS 1-7-515 requires Colorado to conduct risk-limiting audits beginning in 2017. The first risk-limiting election audits under this statute were conducted in November, 2017; the second were conducted in July, 2018.[6] Counties cannot audit contests that cross jurisdictional boundaries (*cross-jurisdictional* contests, such as gubernatorial contests and most federal contests) on their own: margins and risk limits apply to entire contests, not to the portion of a contest included in a county. Colorado has not yet conducted an RLA of a cross-jurisdictional contest, although it has performed RLA-like procedures on individual jurisdictions' portions of some cross-jurisdictional contests. To audit statewide elections and contests that cross county lines, Colorado will need to implement new approaches and make some changes to its auditing software, RLATool.

Colorado's voting systems are heterogeneous. Some counties (containing about 98% of active voters, as of this writing), have voting systems that export cast vote records (CVRs) in a way that the paper ballot corresponding to each CVR can be identified uniquely and retrieved. We call counties with such voting systems *CVR counties*. In CVR counties, auditors can manually check the accuracy of the voting system's interpretation of individual ballots. In other counties (*legacy* or *no-CVR* counties) there is no way to check the accuracy of the system's interpretation of voter intent for individual ballots.

Contests entirely contained in CVR counties can be audited using *ballot-level comparison audits* [6], which compare CVRs to the auditors' interpretation of voter intent directly from paper ballots. Ballot-level comparison audits are currently the most efficient approach to risk-limiting audits in that they require examining fewer ballots than other methods do, when the outcome of the contest under audit is in fact correct. Contests involving no-CVR counties can be audited using *ballot-polling audits* [5,6], which generally require examining more ballots than ballot-level comparison audits to attain the same risk limit.

Colorado's challenge is to audit contests that include ballots cast in both CVR counties and no-CVR counties. There is no literature on how to combine ballot-polling with ballot-level comparisons to audit cross-jurisdictional contests that include voters in CVR counties and voters in no-CVR counties.[7]

Colorado could simply revert to ballot-polling audits for cross-jurisdictional contests that include votes in no-CVR counties, but that would entail a loss of efficiency. Alternatively, Colorado could use batch-level comparison audits, with single-ballot batches in CVR counties and larger batches in no-CVR counties.[8] The statistical theory for such audits has been worked out (see, e.g., [9,10,12,14]

[6] See https://www.sos.state.co.us/pubs/elections/RLA/2017RLABackground.html

[7] See [8] for a different (Bayesian) approach to auditing contests that include both CVR counties and no-CVR counties. In general, Bayesian audits are not risk-limiting.

[8] Since so few ballots are cast in no-CVR counties, cruder approaches might work, for instance, pretending that no-CVR counties had CVRs, but treating any ballot sampled from a no-CVR county as if it had a 2-vote overstatement error. See [1].

and Section A, below); indeed, this is the method that was used in several of California's pilot audits, including the audit in Orange County, California. However, batch-level comparison audits were found to be less efficient than ballot-polling audits in these pilots [2].

Moreover, to use batch-level comparison audits in Colorado would require major changes to RLATool, for reporting batch-level contest results prior to the audit, for drawing the sample, for reporting audit findings, and for determining when the audit can stop. The changes would include modifying data structures, data uploads, random sampling procedures, and the county user interface. No-CVR counties would also have to revise their audit procedures. Among other things, they would need to report vote subtotals for physically identifiable groups of ballots before the audit starts. No-CVR counties with voting systems that can only report subtotals by precinct might have to make major changes to how they handle ballots, for instance, sorting all ballots by precinct. These are large changes.

We show here that SUITE makes possible a "hybrid" RLA that keeps the advantages of ballot-level comparison audits in CVR counties but does not require major changes to how no-CVR counties audit, nor major changes to RLATool. The key is to use stratified sampling with two strata: ballots cast in CVR counties and those cast in no-CVR counties.

In order to use Equation 1, we must develop stratum-level tests for the overstatement error that are appropriate for the corresponding voting system. Sections 3.1 and 3.2 describe these tests for overstatement in the CVR and no-CVR strata, respectively.

3.1 Comparison audits of overstatement quotas

To use comparison auditing in the approach to stratification described above requires extending previous work to test whether the overstatement error exceeds $\lambda_s V_{w\ell}$, rather than simply $V_{w\ell}$. Appendix A derives this generalization for arbitrary batch sizes, including batches consisting of one ballot. The derivation considers only a single contest, but the MACRO test statistic [10,14] automatically extends the result to auditing any number of contests simultaneously. The derivation is for plurality contests, including "vote-for-k" plurality contests. Majority and super-majority contests are a minor modification [9].[9]

3.2 Ballot-polling audits of a tolerable overstatement in votes

To use the new stratification method with ballot-polling requires a different approach than [5] took: their approach tests whether w got a larger *share* of the

[9] So are some forms of preferential and approval voting, such as Borda count, and proportional representation contests, such as D'Hondt [15]. For a derivation of ballot-level comparison risk-limiting audits for super-majority contests, see https://github.com/pbstark/S157F17/blob/master/audit.ipynb. (Last visited 14 May 2018.) Changes for IRV/STV are more complicated.

votes than ℓ, but we need to test whether the margin in votes in the stratum exceeds a threshold (namely, $\lambda_s V_{w\ell}$). This introduces a nuisance parameter, the number of ballots with votes for either w or ℓ. We address this by maximizing the probability ratio in Wald's Sequential Probability Ratio Test [17] over all possible values of the nuisance parameter. Appendix B develops the test.

4 Numerical examples

Jupyter notebooks containing calculations for hybrid stratified audits intended to be relevant for Colorado are available at https://www.github.com/pbstark/CORLA18.

`hybrid-audit-example-1` contains two hypothetical examples. The first has $110,000$ cast ballots, of which 9.1% were in no-CVR counties. The *diluted margin* (the margin in votes, divided by the total number of ballots cast) is 1.8%. In 87% of 10,000 simulations in which the reported results were correct, drawing 500 ballots from the CVR stratum and 700 ballots from the no-CVR stratum (1,200 ballots in all) allowed SUITE to confirm the outcome at 10% risk. For the remaining 13%, further expansion of the audits would have been necessary.

If it were possible to conduct a ballot-level comparison audit for the entire contest, an RLA with risk limit 10% could terminate after examining 263 ballots if it found no errors. A ballot-polling audit of the entire contest would have been expected to examine about 14,000 ballots, more than 10% of ballots cast. The hybrid audit is less efficient than a ballot-level comparison audit, but far more efficient than a ballot-polling audit.

The second contest has 2 million cast ballots, of which 5% were cast in no-CVR counties. The diluted margin is about 20%. The workload for SUITE at 5% risk is quite low: In 100% of 10,000 simulations in which the reported results were correct, auditing 43 ballots from the CVR stratum and 15 ballots from the no-CVR stratum would have confirmed the outcome. If it were possible to conduct a ballot-level comparison audit for the entire contest, an RLA at risk limit 5% could terminate after examining 31 ballots if it found no errors. The additional work for the hybrid stratified audit is disproportionately in the no-CVR counties.

A second notebook, `hybrid-audit-example-2`, illustrates the workflow for SUITE for an election with 2 million ballots cast. The reported margin is just over 1%, but the reported winner and reported loser are actually tied in both strata. The risk limit is 5%. For a sample of 500 ballots from the CVR stratum and 1000 ballots from the no-CVR stratum, the maximum combined P-value is over 25%, so the audit cannot stop there.

A third notebook, `fisher_combined_pvalue`, illustrates the numerical methods used to check whether the maximum combined P-value is below the risk limit. It includes code for the tests in the two strata, for the lower and upper bounds λ_- and λ_+ for λ, for evaluating Fisher's combining function on a grid, and for computing bounds on the P-value via Equation 3.

5 Discussion

We present SUITE, a new class of procedures for RLAs based on stratified random sampling. SUITE is agnostic about the capability of voting equipment in different strata, unlike previous methods, which require batch-level comparisons in every stratum. SUITE allows arbitrary tests to be used in different strata; if those tests are sequentially valid, then the overall RLA is sequential. (Otherwise, multiplicity adjustments might be needed if one wants an audit that escalates in stages. See [9,11] for two approaches.)

Like other RLA methods, SUITE poses auditing as a hypothesis test. The null hypothesis is a union over all partitions of outcome-changing error across strata. The hypothesis is rejected if the maximum P-value over all such partitions is sufficiently small. Each possible partition yields an intersection hypothesis, tested by combining P-values from different strata using Fisher's combining function (or a suitable replacement).

Among other things, the new approach solves a current problem in Colorado: how to conduct RLAs of contests that cross jurisdictional lines, such as statewide contests and many federal contests.

We give numerical examples in Jupyter notebooks that can be modified to estimate the workload for different contest sizes, margins, and risk limits. In our numerical experiments, the new method requires auditing far fewer ballots than previous approaches would.

A Comparison tests for an overstatement quota

A.1 Notation

- $\mathcal{W}$: the set of reported winners of the contest
- $\mathcal{L}$: the set of reported losers of the contest
- N_s ballots were cast in stratum s. (The contest might not appear on all N_s ballots.)
- P "batches" of ballots are in stratum s. A batch contains one or more ballots. Every ballot in stratum s is in exactly one batch.
- n_p: number of ballots in batch p. $N_s = \sum_{p=1}^{P} n_p$.
- $v_{pi} \in \{0, 1\}$: reported votes for candidate i in batch p
- $a_{pi} \in \{0, 1\}$: actual votes for candidate i in batch p. If the contest does not appear on any ballot in batch p, then $a_{pi} = 0$.
- $V_{w\ell,s} \equiv \sum_{p=1}^{P}(v_{pw} - v_{p\ell})$: Reported margin in stratum s of reported winner $w \in \mathcal{W}$ over reported loser $\ell \in \mathcal{L}$, in votes.
- $V_{w\ell}$: overall reported margin in votes of reported winner $w \in \mathcal{W}$ over reported loser $\ell \in \mathcal{L}$ for the entire contest (not just stratum s)
- $V \equiv \min_{w\in\mathcal{W},\ell\in\mathcal{L}} V_{w\ell}$: smallest reported overall margin in votes between any reported winner and reported loser
- $A_{w\ell,s} \equiv \sum_{p=1}^{P}(a_{pw} - a_{p\ell})$: actual margin in votes in the stratum of reported winner $w \in \mathcal{W}$ over reported loser $\ell \in \mathcal{L}$
- $A_{w\ell}$: actual margin in votes of reported winner $w \in \mathcal{W}$ over reported loser $\ell \in \mathcal{L}$ for the entire contest (not just in stratum s)

A.2 Reduction to maximum relative overstatement

If the contest is entirely contained in stratum s, then the reported winners of the contest are the actual winners if

$$\min_{w \in \mathcal{W}, \ell \in \mathcal{L}} A_{w\ell,s} > 0.$$

Here, we address the case that the contest may include a portion outside the stratum. To combine independent samples in different strata, it is convenient to be able to test whether the net overstatement error in a stratum exceeds a given threshold.

Instead of testing that condition directly, we will test a condition that is sufficient but not necessary for the inequality to hold, to get a computationally simple test that is still conservative (i.e., the level is not larger than its nominal value).

For every winner, loser pair (w, ℓ), we want to test whether the overstatement error exceeds some threshold, generally one tied to the reported margin between w and ℓ. For instance, for a hybrid stratified audit, we set the threshold to be $\lambda_s V_{w\ell}$.

We want to test whether

$$\sum_{p=1}^{P} (v_{pw} - a_{pw} - v_{p\ell} + a_{p\ell})/V_{w\ell} \geq \lambda_s.$$

The maximum of sums is not larger than the sum of the maxima; that is,

$$\max_{w \in \mathcal{W}, \ell \in \mathcal{L}} \sum_{p=1}^{P} (v_{pw} - a_{pw} - v_{p\ell} + a_{p\ell})/V_{w\ell} \leq \sum_{p=1}^{P} \max_{w \in \mathcal{W}, \ell \in \mathcal{L}} (v_{pw} - a_{pw} - v_{p\ell} + a_{p\ell})/V_{w\ell}.$$

Define

$$e_p \equiv \max_{w \in \mathcal{W} \ell \in \mathcal{L}} (v_{pw} - a_{pw} - v_{p\ell} + a_{p\ell})/V_{w\ell}.$$

Then no reported margin is overstated by a fraction λ_s or more if

$$E \equiv \sum_{p=1}^{P} e_p < \lambda_s.$$

Thus if we can reject the hypothesis $E \geq \lambda_s$, we can conclude that no pairwise margin was overstated by as much as a fraction λ_s.

Testing whether $E \geq \lambda_s$ would require a very large sample if we knew nothing at all about e_p without auditing batch p: a single large value of e_p could make E arbitrarily large. But there is an *a priori* upper bound for e_p. Whatever the reported votes v_{pi} are in batch p, we can find the potential values of the actual votes a_{pi} that would make the error e_p largest, because a_{pi} must be between 0 and n_p, the number of ballots in batch p:

$$\frac{v_{pw} - a_{pw} - v_{p\ell} + a_{p\ell}}{V_{w\ell}} \leq \frac{v_{pw} - 0 - v_{p\ell} + n_p}{V_{w\ell}}.$$

Hence,

$$e_p \le \max_{w \in \mathcal{W}, \ell \in \mathcal{L}} \frac{v_{pw} - v_{p\ell} + n_p}{V_{w\ell}} \equiv u_p. \tag{4}$$

Knowing that $e_p \le u_p$ might let us conclude reliably that $E < \lambda_s$ by examining only a small number of batches—depending on the values $\{u_p\}_{p=1}^{P}$ and on the values of $\{e_p\}$ for the audited batches.

To make inferences about E, it is helpful to work with the *taint* $t_p \equiv \frac{e_p}{u_p} \le 1$. Define $U \equiv \sum_{p=1}^{P} u_p$. Suppose we draw batches at random with replacement, with probability u_p/U of drawing batch p in each draw, $p = 1, \ldots, P$. (Since $u_p \ge 0$, these are all positive numbers, and they sum to 1, so they define a probability distribution on the P batches.)

Let T_j be the value of t_p for the batch p selected in the jth draw. Then $\{T_j\}_{j=1}^{n}$ are IID, $\mathbb{P}\{T_j \le 1\} = 1$, and

$$\mathbb{E}T_1 = \sum_{p=1}^{P} \frac{u_p}{U} t_p = \frac{1}{U} \sum_{p=1}^{P} u_p \frac{e_p}{u_p} = \frac{1}{U} \sum_{p=1}^{P} e_p = E/U.$$

Thus $E = U\mathbb{E}T_1$. So, if we have strong evidence that $\mathbb{E}T_1 < \lambda_s/U$, we have strong evidence that $E < \lambda_s$.

This approach can be simplified even further by noting that u_p has a simple upper bound that does not depend on v_{pi}. At worst, the reported result for batch p shows n_p votes for the "least-winning" apparent winner of the contest with the smallest margin, but a hand interpretation would show that all n_p ballots in the batch had votes for the runner-up in that contest. Since $V_{w\ell} \ge V \equiv \min_{w \in \mathcal{W}, \ell \in \mathcal{L}} V_{w\ell}$ and $0 \le v_{pi} \le n_p$,

$$u_p = \max_{w \in \mathcal{W}, \ell \in \mathcal{L}} \frac{v_{pw} - v_{p\ell} + n_p}{V_{w\ell}} \le \max_{w \in \mathcal{W}, \ell \in \mathcal{L}} \frac{n_p - 0 + n_p}{V_{w\ell}} \le \frac{2n_p}{V}.$$

Thus if we use $2n_p/V$ in lieu of u_p, we still get conservative results. (We also need to re-define U to be the sum of those upper bounds.) An intermediate, still conservative approach would be to use this upper bound for batches that consist of a single ballot, but use the sharper bound (4) when $n_p > 1$. Regardless, for the new definition of u_p and U, $\{T_j\}_{j=1}^{n}$ are IID, $\mathbb{P}\{T_j \le 1\} = 1$, and

$$\mathbb{E}T_1 = \sum_{p=1}^{P} \frac{u_p}{U} t_p = \frac{1}{U} \sum_{p=1}^{P} u_p \frac{e_p}{u_p} = \frac{1}{U} \sum_{p=1}^{P} e_p = E/U.$$

So, if we have evidence that $\mathbb{E}T_1 < \lambda_s/U$, we have evidence that $E < \lambda_s$.

A.3 Testing $\mathbb{E}T_1 \ge \lambda_s/U$

A variety of methods are available to test whether $\mathbb{E}T_1 < \lambda_s/U$. One particularly elegant sequential method is based on Wald's Sequential Probability Ratio Test (SPRT) [17]. Harold Kaplan pointed out this method on a website that no longer exists. A derivation of this *Kaplan-Wald* method is in Appendix A of [15]; to apply the method here, take $t = \lambda_s$ in their equation 18. A different sequential method, the *Kaplan-Markov* method (also due to Harold Kaplan), is given in [12].

B Ballot-polling tests for an overstatement quota

In this section, we derive a ballot-polling test of the hypothesis that the margin (in votes) in a single stratum exceeds a threshold c.

B.1 Wald's SPRT with a nuisance parameter

Consider a single stratum s containing N_s ballots, of which $N_{w,s}$ have a vote for w but not for ℓ, $N_{\ell,s}$ have a vote for ℓ but not for w, and $N_{u,s} = N_s - N_{w,s} - N_{\ell,s}$ have votes for both w and ℓ or neither w nor ℓ, including undervotes and invalid ballots. Ballots are drawn sequentially without replacement, with equal probability of selecting each as-yet-unselected ballot in each draw.

We want to test the compound hypothesis that $N_{w,s} - N_{\ell,s} \le c$ against the alternative that $N_{w,s} = V_{w,s}$, $N_{\ell,s} = V_{\ell,s}$, and $N_{u,s} = V_{u,s}$, with $V_{w,s} - V_{\ell,s} > c$.

The values $V_{w,s}$, $V_{\ell,s}$, and $V_{u,s}$ are the reported results for stratum s (or values related to those reported results; see [5]). In this problem, $N_{u,s}$ (equivalently, $N_{w,s} + N_{\ell,s}$) is a nuisance parameter: we care about $N_{w,s} - N_{\ell,s}$.

Let X_k be w, ℓ, or u according to whether the ballot selected on the kth draw shows a vote for w but not ℓ, ℓ but not w, or something else. Let $W_n \equiv \sum_{k=1}^{n} 1_{X_k = w}$; and define L_n and U_n analogously.

The probability of a given data sequence $X_1, \ldots, X_n$ under the alternative hypothesis is

$$\frac{\prod_{i=0}^{W_n-1}(V_{w,s} - i) \ \prod_{i=0}^{L_n-1}(V_{\ell,s} - i) \ \prod_{i=0}^{U_n-1}(V_{u,s} - i)}{\prod_{i=0}^{n-1}(N_s - i)}.$$

If $L_n \ge W_n - cn/N_s$, the data obviously do not provide evidence against the null, so we suppose that $L_n < W_n - cn/N_s$, in which case, the element of the null that will maximize the probability of the observed data has $N_{w,s} - c = N_{\ell,s}$. Under the null hypothesis, the probability of $X_1, \ldots, X_n$ is

$$\frac{\prod_{i=0}^{W_n-1}(N_{w,s} - i) \ \prod_{i=0}^{L_n-1}(N_{w,s} - c - i) \prod_{i=0}^{U_n-1}(N_{u,s} - i)}{\prod_{i=0}^{n}(N_s - i)},$$

for some value $N_{w,s}$ and the corresponding $N_{u,s} = N_s - 2N_{w,s} + c$. How large can that probability be under the null? The probability under the null is maximized by any integer $x \in \{\max(W_n, L_n + c), \ldots, (N - U_n)/2\}$ that maximizes

$$\prod_{i=0}^{W_n-1} (x - i) \ \prod_{i=0}^{L_n-1} (x - c - i) \ \prod_{i=0}^{U_n-1} (N_s - 2x + c - i).$$

The logarithm is monotonic, so any maximizer x^* also maximizes

$$f(x) = \sum_{i=0}^{W_n-1} \ln(x-i) + \sum_{i=0}^{L_n-1} \ln(x-c-i) + \sum_{i=0}^{U_n-1} \ln(N_s - 2x + c - i).$$

The first two terms on the right increase monotonically with x and the last term decreases monotonically with x. This yields bounds without having to evaluate f everywhere. Suppose $y < z$. Then for all integer x between y and z,

$$f(x) \leq \sum_{i=0}^{W_n-1} \ln(z-i) + \sum_{i=0}^{L_n-1} \ln(z-c-i) + \sum_{i=0}^{U_n-1} \ln(N_s - 2y + c - i).$$

The optimization problem can be solved using a branch and bound approach. For instance, start by evaluating

$$f^+(x) \equiv \sum_{i=0}^{W_n-1} \ln(x-i) + \sum_{i=0}^{L_n-1} \ln(x-c-i)$$

and

$$f^-(x) \equiv \sum_{i=0}^{U_n-1} \ln(N_s - 2x + c - i)$$

at $\max(W_n, L_n + c)$, $(N_s - U_n)/2$, and their midpoint, to get the values of $f = f^+ + f^-$ at those three points, along with upper bounds on f on the ranges between them. At stage j, we have evaluated f, f^+, and f^- at j points $x_1 < x_2 < \ldots < x_j$, and we have upper bounds on f on the $j-1$ ranges $R_m = \{x_m, x_m + 1, \ldots, x_{m+1}\}$ between those points. Let U_m be the upper bound on $f(x)$ for $x \in R_m$. Suppose that for some h, $f(x_h) = \max_{m=1}^{j} U_m$. Then $x^* = x_h$ is a global maximizer of f. If there is some $U_m > \max_i f(x_i)$, then subdivide the range with the largest U_m, calculate f, f^+, and f^- at the new point, and repeat. This algorithm must terminate by identifying a global maximizer x^* after a finite number of steps.

A conservative P-value for the null hypothesis after n items have been drawn is thus

$$P_n = \frac{\prod_{i=0}^{W_n-1}(x^* - i) \prod_{i=0}^{L_n-1}(x^* - c - i) \prod_{i=0}^{U_n-1}(N_s - 2x^* + c - i)}{\prod_{i=0}^{W_n-1}(V_{w,s} - i) \prod_{i=0}^{L_n-1}(V_{\ell,s} - i) \prod_{i=0}^{U_n-1}(V_{u,s} - i)}.$$

Because the test is built on Wald's SPRT, the sample can expand sequentially and (if the null hypothesis is true) the chance that $P_n < p$ is never larger than p. That is, $\Pr\{\inf_n P_n < p\} \leq p$ if the null is true.

A Jupyter notebook implementing this approach is given in https://github.com/pbstark/CORLA18.

References

1. Bañuelos, J., Stark, P.: Limiting risk by turning manifest phantoms into evil zombies. Tech. rep., arXiv.org (2012), http://arxiv.org/abs/1207.3413, retrieved 17 July 2012
2. California Secretary of State: California Secretary of State Post-Election Risk-Limiting Audit Pilot Program 2011-2013: Final Report to the United States Election Assistance Commission. http://votingsystems.cdn.sos.ca.gov/oversight/risk-pilot/final-report-073014.pdf Retrieved 6 May 2018 (2014)
3. Grimmett, G.R., Stirzaker, D.R.: Probability and Random Processes. Oxford University Press (August 2001), http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&amp;path=ASIN/0198572220
4. Higgins, M., Rivest, R., Stark, P.: Sharper p-values for stratified post-election audits. Statistics, Politics, and Policy **2**(1) (2011), http://www.bepress.com/spp/vol2/iss1/7
5. Lindeman, M., Stark, P., Yates, V.: BRAVO: Ballot-polling risk-limiting audits to verify outcomes. In: Proceedings of the 2011 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections (EVT/WOTE '11). USENIX (2012)
6. Lindeman, M., Stark, P.B.: A gentle introduction to risk-limiting audits. IEEE Security and Privacy **10**, 42–49 (2012)
7. Pesarin, F., Salmaso, L.: Permutation tests for complex data: Theory, applications, and software. John Wiley and Sons, Ltd., West Sussex, UK (2010)
8. Rivest, R.L.: Bayesian tabulation audits: Explained and extended (January 1, 2018), https://arxiv.org/abs/1801.00528
9. Stark, P.: Conservative statistical post-election audits. Ann. Appl. Stat. **2**, 550–581 (2008), http://arxiv.org/abs/0807.4005
10. Stark, P.: Auditing a collection of races simultaneously. Tech. rep., arXiv.org (2009), http://arxiv.org/abs/0905.1422v1
11. Stark, P.: CAST: Canvass audits by sampling and testing. IEEE Transactions on Information Forensics and Security, Special Issue on Electronic Voting **4**, 708–717 (2009)
12. Stark, P.: Risk-limiting post-election audits: P-values from common probability inequalities. IEEE Transactions on Information Forensics and Security **4**, 1005–1014 (2009)
13. Stark, P.: Risk-limiting vote-tabulation audits: The importance of cluster size. Chance **23**(3), 9–12 (2010)
14. Stark, P.: Super-simple simultaneous single-ballot risk-limiting audits. In: Proceedings of the 2010 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections (EVT/WOTE '10). USENIX (2010), http://www.usenix.org/events/evtwote10/tech/full_papers/Stark.pdf
15. Stark, P.B., Teague, V.: Verifiable european elections: Risk-limiting audits for d'hondt and its relatives. JETS: USENIX Journal of Election Technology and Systems **3.1** (2014), https://www.usenix.org/jets/issues/0301/stark
16. Stark, P.B., Wagner, D.A.: Evidence-based elections. IEEE Security and Privacy **10**, 33–41 (2012)
17. Wald, A.: Sequential tests of statistical hypotheses. Ann. Math. Stat. **16**, 117–186 (1945)

Computing the Margin of Victory in Preferential Parliamentary elections

Michelle Blom, Peter J. Stuckey, and Vanessa J. Teague

[michelle.blom,p.stuckey,vjteague]@unimelb.edu.au
School of Computing and Information Systems
The University of Melbourne
Parkville, Australia

Abstract. We show how to use automated computation of election margins to assess the number of votes that would need to change in order to alter a parliamentary outcome for single-member preferential electorates. In the context of increasing automation of Australian electoral processes, and accusations of deliberate interference in elections in Europe and the USA, this work forms the basis of a rigorous statistical audit of the parliamentary election outcome. Our example is the New South Wales Legislative Council election of 2015, but the same process could be used for any similar parliament for which data was available, such as the Australian House of Representatives.

1 Introduction

The party that wins a majority of seats in a parliamentary election may not be the party that wins a majority of votes. This has been examined extensively in the United States [7, 8]. In Australian parliamentary elections, even the notion of a "popular majority" is poorly defined because Australian voters rank their candidates in order of preference. But similar results occur: sometimes in practice the Parliamentary winner is not the popular majority winner and there are even some systematic biases [2]. Nevertheless it is often assumed by the public and the media that a party that wins a comfortable overall margin will comfortably win the parliamentary election. Of course, this is not necessarily true.

In this paper we focus on computing the Parliamentary election margin: the minimal number of votes that need to be changed, in a particular election outcome, to switch the Parliamentary winner. This may be much less than the margin between the popular votes of the two major parties.

There are two ways that an Australian parliamentary election may be closer than it seems. First, there may be many seats held by a very small margin. Second, even within one seat, the margin may be smaller than it appears. Australia's preferential voting system proceeds by iteratively eliminating candidates until only two remain, then selecting the one with a larger tally of votes. A naive observer might think that the margin of victory is the number of votes that need to be switched to reverse the winner in this last step (*i.e.* half the difference in the final tallies)—we call this the *last-round margin*. The true margin may be much smaller, however, as changing an early elimination step may cascade into a completely different elimination order. Computing the correct margin for preferential voting is, in general, a computationally difficult problem, but an efficient solution has been demonstrated [1].

In earlier work, Blom *et al.* [1] present an algorithm for computing the margin of victory in Instant Runoff Voting (IRV) elections (also commonly referred to as Alternative Vote elections). In an Australian state or federal Parliamentary election, an IRV election is held in each of a number of districts, electing a single candidate to a seat in the lower house. The party (or coalition of parties) that holds the majority of seats in the lower house, wins the election. Recall that Australian voters rank candidates in order of preference (for example, the ranking $[a, c, b]$ expresses a first preference for candidate a, a second for c, and a third for b). A change to a vote replaces its ranking over candidates with an alternate ranking (for example, replacing ranking $[a, c, b]$ with $[c, b, a]$). In this paper, we are interested in computing the smallest number of votes (of those cast) that need to be changed to ensure that a different party (or coalition of parties) wins the majority of seats, or that no party (or specific coalition of parties) wins a majority of seats (leading to a hung parliament). Computing the Parliamentary election margin requires a slight modification to the algorithm of [1], in that we must compute the margin of victory *with respect to a specific set of alternate winners* in each seat.

For example, to determine how many votes we would need to change to ensure that Labor wins a majority of seats (in place of the Liberal/National coalition), we would look at manipulations in which seats won by non-Labor candidates are consequently awarded to the Labor candidate.[1] A process of sorting the seats in increasing order of margin, and adding the margins in the necessary number of seats yields the desired Parliamentary election margin.

As a case study, we use data from the 2015 NSW state election to compute the margin by which the Liberal/National coalition won. The popular margin was high—the Liberal/National coalition won 46% of formal first-preference votes compared with 34% for the Labor parties and 10% for The Greens.[2] The coalition won 54 seats compared to Labor's 34. We find, however, that the number of votes necessary to switch the parliamentary outcome is less than 0.1%.

In prior work on US IRV elections, Blom *et al.* [1] found that the true margin is almost always the last-round margin, though exceptions did occur. This is also true of the NSW 2015 election where, for example, the Lismore seat has a last-round margin of 1173, but the true margin of victory is only 209 votes.

The source code used to compute our results is located at:

```
https://github.com/michelleblom/margin-irv
```

These techniques could be easily applied to any parliamentary outcome for which complete vote data was available. This analysis could become standard procedure for any parliamentary election with automated ballot scanning.

[1] The Liberal, National, and Labor parties are three Australian political parties.

[2] This is from http://pastvtr.elections.nsw.gov.au/SGE2015/la/state/formal/index.htm

1.1 Notation

Below we give common three letter codes used to refer to parties in the 2015 New South Wales (NSW) state election.

LAB	Australian Labor Party
CLP	Country Labor Party
LIB	Liberal Party of Australia
NAT	National Party of Australia
GRN	Australian Greens
IND	Independent (belonging to no party)

1.2 Summary of Results

Of the 4.56 million votes cast in the 2015 New South Wales state election, we have determined that it would have taken:

- 22,746 vote changes for the Labor/Country Labor party to gain the 13 additional seats they need to win government (with 47 seats),
- 16,349 vote changes for a Labor/Greens coalition to gain the 10 additional seats they need to win government, and
- 10,398 vote changes to lose the Liberal/National coalition 8 seats and hence produce a hung parliament.

1.3 Auditing and Accuracy Testing in Elections

The margin computation tools presented in this paper can be used, whenever data[3] is available, to check automatically whether a known problem in an election was large enough to change the outcome. Similarly, when a known number of votes were received over an insecure or unscrutinisable channel, this could be used to decide whether that might have been enough to alter the outcome.

Conversely, it could be used to generate evidence that the election outcome is right.

These calculations could be used as the basis for a rigorous risk-limiting audit to confirm (or overturn) the announced election outcome. Risk limiting audits [4] take an iterative random sample of the paper ballots to check how well they reflect the announced outcome. An audit has *risk-limit* α if a mistaken outcome is guaranteed to be detected with a probability of at least $1 - \alpha$. Either the audit concludes with a certain confidence that the outcome is right, or it finds so many errors that a full manual recount is warranted. The audit process is parameterised by the margin of victory in the election. Kroll *et al.* [3] have devised audits for parliamentary outcomes but, like most US research, they focus on simple first-past-the-post elections in which the margin is obvious.

This is particularly important now that the Australian Parliament's Joint Standing Committee on Electoral Matters has recommended automated scanning of the ballot

[3] An electronic record of the preferences expressed in each paper ballot, after scanning and digitisation

```
Initially, all candidates remain standing (are not eliminated)
While there is more than one candidate standing
    For every candidate $c$ standing
        Tally (count) the votes in which $c$ is the highest-ranked
        candidate of those standing
    Eliminate the candidate with the smallest tally
The winner is the one candidate not eliminated
```

Fig. 1: The IRV counting algorithm: the candidate with the smallest tally is repeatedly eliminated, with the ballots in their tally redistributed to remaining candidates according to their next preference.

papers [6]. The overall Parliamentary margin could be quickly calculated using our methods. Rigorous risk-limiting audits could then be performed for each electorate, immediately after the election, in order to provide evidence that the overall election outcome was correct.

In a time where outside influencing of elections is a constant source of news, and where more and more elections systems involve electronic systems, either for voting or counting votes, it is critical that we have mechanisms in place to generate evidence of accurate election results, and indeed to check what degree of manipulation must have taken place for the election result to have been altered.

2 Background

The lower houses of parliaments in the Australian federal and state elections are the result of a number of independent Instant Runoff Voting (IRV) elections for a set of single-member electorates (seats). Each seat has a number of candidates, and each vote consists of an ordered list of the candidates for that seat.[4]

The tallying of votes in an IRV election proceeds by a series of rounds in which the candidate with the lowest number of votes is eliminated (see Figure 1) with the last remaining candidate declared the winner. All votes in an eliminated candidate's tally are distributed to the next most-preferred (remaining) candidate in their ranking.

Let $\mathcal{C}$ be the set of candidates in an IRV election $\mathcal{B}$. We refer to sequences of candidates π in list notation (e.g., $\pi = [c_1, c_2, c_3, c_4]$), and use such sequences to represent both votes and elimination orders. We will often treat a sequence as the set of elements it contains. An election $\mathcal{B}$ is defined as a multiset[5] of votes, each vote $b \in \mathcal{B}$ a sequence of candidates in $\mathcal{C}$, with no duplicates, listed in order of preference (most preferred to least preferred). Let $\mathit{first}(\pi)$ denote the first candidate appearing in sequence π (e.g., $\mathit{first}([c_2, c_3]) = c_2$). In each round of vote counting, there are a current set of eliminated candidates $\mathcal{E}$ and a current set of candidates still standing $\mathcal{S} = \mathcal{C} \setminus \mathcal{E}$. The winner c_w of the election is the last standing candidate.

[4] Most Australian elections require all preferences to be filled in, but some allow partial lists or several equal-last candidates. Our analysis extends to all these cases.

[5] A multiset allows for the inclusion of duplicate items.

Each candidate $c \in \mathcal{C}$ has a *tally* of votes. Votes are added to this tally upon the elimination of a candidate $c' \in \mathcal{C} \setminus \{c\}$, and are redistributed from this tally upon the elimination of c.

Definition 1. ***Tally*** $\mathbf{t}_{\mathcal{S}}(\mathbf{c})$ *Given candidates $\mathcal{S} \subseteq \mathcal{C}$ are still standing in an election $\mathcal{B}$, the tally for candidate $c \in \mathcal{C}$, denoted $t_{\mathcal{S}}(c)$, is defined as the number of votes $b \in \mathcal{B}$ for which c is the most-preferred candidate of those remaining. Let $p_{\mathcal{S}}(b)$ denote the sequence of candidates mentioned in b that are also in $\mathcal{S}$.*

$$t_{\mathcal{S}}(c) = \ | \, [b \mid b \in \mathcal{B}, c = \mathit{first}(p_{\mathcal{S}}(b))] \, | \tag{1}$$

Definition 2. ***Margin of Victory (MOV)*** *The MOV in an election with candidates $\mathcal{C}$ and winner $c_w \in \mathcal{C}$, is the smallest number of votes whose ranking must be modified (by an adversary) so that a candidate $c' \in \mathcal{C} \setminus \{c_w\}$ is elected.*

Often the last round margin (LRM) is used as a proxy for the margin of victory.

Definition 3. ***Last Round Margin (****LRM****)*** *The LRM of an election, in which two candidates $\mathcal{S} = \{c, c'\}$ remain with $t_{\mathcal{S}}(c)$ and $t_{\mathcal{S}}(c')$ votes in their tallies, is equal to half the difference between the tallies of c and c' rounded up.*

$$LRM = \lceil \frac{|t_{\mathcal{S}}(c) - t_{\mathcal{S}}(c')|}{2} \rceil \tag{2}$$

In this paper, we are interested in a more restricted version of margin of victory, which is the margin of victory over a subset of the non-winning candidates.

Definition 4. ***Margin of Victory over Candidates*** $\mathcal{A}$ ***(MOVC)*** *The MOVC in an election with candidates $\mathcal{C}$ and winner $c_w \in \mathcal{C}$ over the alternate candidates $\mathcal{A} \subseteq \mathcal{C} \setminus \{c_w\}$, is the smallest number of votes whose ranking must be modified (by an adversary) so that a candidate $c' \in \mathcal{A}$ is elected.*

While the MOV calculates the number of votes required to be changed to alter the winner, the MOVC calculates the number of votes required to be changed to alter the winner *to one of a set $\mathcal{A}$*. We will require this finer information in order to calculate the smallest number of votes for a different party or coalition to win the election.

Example 1. Consider an election between candidates a, b, and c with the election profile shown in Table 1. The initial tallies of a, b, and c are 55, 41, and 40 votes, respectively, hence c is eliminated. Candidates a and b consequently have tallies of 80 and 41 votes, giving a the victory with a last round margin of 20 votes. Consider changing 1 of the [b, c] votes to a [c] vote. Then the initial tallies are $\{a : 55, b : 40, c : 41\}$ and b is eliminated. Candidates a and c consequently have tallies of 55 and 81 votes, and c is the winner of the election.

Clearly the MOV is 1 vote. The MOVC for $\{b\}$ is 10, which is achieved by changing 5 votes from [a] to [b, c] and 5 from [a] to [c], giving first round tallies of $\{a : 45, b : 46, c : 45\}$. An adversary can choose to eliminate a leaving b and c with tallies of 46 and 45 votes, and b winning the election. □

Ranking	Count
[a]	55
[c, a]	25
[b, c]	41
[c]	15

(a)

Candidate	Round 1	Round 2
a	55	80
b	41	41
c	40	—

(b)

Table 1: IRV example, with (a) the number of votes cast with each listed ranking over candidates a, b, c, and (b) tallies after each round of vote counting.

2.1 Computing margins for an IRV election

Blom *et al.* [1] present a branch-and-bound algorithm (denoted *margin-irv*) for efficiently computing the margin of victory in an IRV election. This algorithm improves upon an existing method by Magrino *et al.* [5].

Given an IRV election with winning candidate c_w, *margin-irv* traverses a tree defining all possible *alternate* orders of candidate elimination (that result in a winning candidate other than c_w). As the algorithm explores these alternate elimination sequences, it solves a mixed integer program (MIP) to determine the minimum number of vote manipulations required to realise each elimination order. The ultimate goal is to find an elimination sequence, in which an alternate winner is elected, that requires the smallest number of vote changes to realise. Searching through the entire space of alternate elimination sequences would be too combinatorially complex, however, and so *margin-irv* incorporates rules for pruning sections of this tree from consideration. The result is an efficient algorithm for computing electoral margins.

A description of both the *margin-irv* algorithm, and the original branch-and-bound method of Magrino *et al.* [5], can be found in Blom *et al.* [1]. We summarise *margin-irv* in this section, and outline how it can be altered to compute a margin over a set of candidates $\mathcal{A}$ (the MOVC). Appendix B provides the full *margin-irv* algorithm for computing the MOVC for a single seat.

Given an IRV election with candidates $\mathcal{C}$ and winner $c_w \in \mathcal{C}$, the *margin-irv* algorithm starts by adding $|\mathcal{C}| - 1$ partial elimination sequences to the search tree, one for each of alternate winner $c'_w \in \mathcal{C} \setminus \{c_w\}$. These partial sequences form a frontier F. Each of these sequences contains a single candidate – the alternate winner in question. Following the basic structure of a branch-and-bound algorithm, we compute, for each partial sequence $\pi \in F$, a lower bound on the number of vote changes required to realise a elimination sequence that *ends* in π. These lower bounds are used to guide construction of the search tree, and are computed by both solving a MIP, and applying several rules for lower bound computation. The partial sequence π with the smallest lower bound is selected and *expanded*. For each candidate $c \in \mathcal{C}$ that is not already present in π, we create a new sequence with c appended to the front. For example, given a set of candidates c_1, c_2, and c_3, with winning candidate c_3, the partial sequence $\pi = [c_2]$ will be expanded to create two new sequences $[c_1, c_2]$ and $[c_3, c_2]$. We evaluate each new sequence π' created by assigning it a lower bound on the number of votes required to realise any elimination order ending in π'.

While exploring and building elimination sequences, *margin-irv* maintains a running *upper bound* on the value of the true margin. This upper bound is initialised to the last round margin of the election. When a sequence π containing all candidates is constructed, our MIP computes the exact number of vote manipulations required to realise it. If this number is lower than our current upper bound, the upper bound is revised, and all orders on our frontier with a lower bound greater than or equal to it are pruned from consideration (removed from our frontier). This process continues until our frontier is empty (we have considered or pruned all possible alternate elimination sequences). The value of the running upper bound is the true margin of victory of the election.

Its easy to extend the *margin-irv* algorithm to also calculate MOVC for a set of alternate winners $\mathcal{A}$. In the first step of the algorithm, rather than adding a node for each alternate winner in $\mathcal{C} \setminus \{c_w\}$ we add a node only for each of the alternate candidates in $\mathcal{A}$. The remainder of the algorithm is unchanged. With this modification, *margin-irv* will only explore alternate election outcomes that result in one of the candidates in $\mathcal{A}$ winning the election.

3 Calculating the Number of Votes to Change a Parliamentary Election Outcome

Given a set S of seats in a parliament, a winning coalition P is a set of parties such that the number of seats won by that coalition is at least some defined threshold T. Usually $T = \lceil \frac{|S|+1}{2} \rceil$, requiring the coalition to win more than half the seats. The NSW Legislative Assembly has 93 seats, and so 47 are required to win government.

We can use this threshold to calculate the number of vote changes required to change a parliamentary election result as follows. Assume the coalition won $W \geq T$ seats. We calculate the MOVC for each seat s won by the coalition P for the set of alternate candidates in that election *not* in coalition P. We then sort the MOVC values, and choose the $W - T + 1$ seats O with the least MOVC values. The sum of the MOVC of these seats O is the number of changes in votes required to remove the victory of the winning coalition P, and hence change the outcome of the election.

Note that if the coalition is a single party $P = \{p\}$, or more generally if no seat has two candidates from the coalition, then the MOVC values required are identical to MOV values. This is the case for the NSW Legislative Election where no seat has both a Liberal (LIB) and National (NAT) candidate. The above procedure examines how we might rob the original winning coalition P of its victory. However, we are interested in computing the number of vote changes required to award victory to a specific party or coalition of parties P' (such as a Labor (LAB)/Greens (GRN) coalition).

We can use a similar approach to calculate the number of vote changes required to change a parliamentary election outcome so that another coalition P' would win instead. Assume P' won $W' < T$ seats. We calculate the MOVC for each seat s not won by coalition P' with the set of alternate candidates $\mathcal{A}$ equal to the set of candidates belonging to parties in P'. We then sort the MOVC values, and choose the $W' - T$ seats O' with the least MOVC values. The sum of the MOVC of these seats O' is the number of changes in votes required to give a parliamentary victory to coalition P'.

Seat	$\lvert\mathcal{C}\rvert$	Last-round margin	True margin	Winner
Lismore	6	1173	209	NAT
Balina	7	1267	1130	GRN
Heffron	5	5835	5824	LAB
Maitland	6	5446	4012	CLP
Willoughby	6	10247	10160	LIB

Table 2: The 5 seats in the 2015 NSW Legislative Assembly Parliamentary Election in which the last-round margin did not equal the true margin of victory.

Seat	$\lvert\mathcal{C}\rvert$	Last-round margin	True margin	Winner
East Hills	5	189	189	LIB
Lismore	6	1173	209	NAT
Upper Hunter	6	866	866	NAT
Monaro	5	1122	1122	NAT
Coogee	5	1243	1243	LIB
Tweed	5	1291	1291	NAT
Penrith	8	2576	2576	LIB
Holsworthy	6	2902	2902	LIB

Table 3: The 8 seats, won by a LIB or NAT candidate, with the lowest MOV.

Again if the coalition P' was always the alternate winner in the calculation of the MOV, then the MOVC and MOV calculations will coincide, and indeed if P' is a strong existing coalition it is likely that it is the alternate winner in most seats with the lowest MOVC.

4 Results

The NSW Legislative Assembly Parliamentary Election of 2015 was contested by major parties: Liberal (LIB), National (NAT), Green (GRN), Labor (LAB) and Country Labor (CLP); as well as a number of minor parties and independents (IND). We found 5 seats in which the true margin was not the last-round margin. These seats are listed in Table 2, alongside the number of candidates up for election in each electorate ($|\mathcal{C}|$), the last-round margin for the seat, the true margin of victory for the seat, and the party whose candidate won the seat.

The LIB/NAT coalition won 54 seats to have a winning majority. In order to lose this majority, they must lose $54 - 47 + 1 = 8$ seats. Since no seat ran both a LIB and a NAT candidate, we can use the MOV values to calculate the number of votes required to lose 8 seats. The 8 LIB/NAT seats with the lowest MOV are listed in Table 3. For Lismore, the MOV differs substantially from the last-round margin. Hence, the total number of votes required for the LIB/NAT coalition to lose their majority is 10,398 (the sum of the ‘True margin’ values in the 4th column of Table 3).

For a LAB and CLP coalition to win the election we need them win to $47 - 34 = 13$ more seats. The 13 seats with the lowest MOVC for a change to LAB/CLP are listed in Table 4.

Seat	$\lvert\mathcal{C}\rvert$	Last-round margin	True margin	Winner	MOVC
East Hills	5	189	189	LIB	189
Lismore	6	1173	209	NAT	209
Upper Hunter	6	866	866	NAT	866
Monaro	5	1122	1122	NAT	1122
Balina	7	1267	1130	GRN	1130
Coogee	5	1243	1243	LIB	1243
Tweed	5	1291	1291	NAT	1291
Balmain	7	1731	1731	GRN	1731
Penrith	8	2576	2576	LIB	2576
Holsworthy	6	2902	2902	LIB	2902
Goulburn	6	2945	2945	LIB	2945
Oatley	5	3006	3006	LIB	3006
Newtown	7	3536	3536	GRN	3536

Table 4: The 13 seats with the lowest MOVC for a change in winner to LAB/CLP.

The total number of votes required to give an LAB/CLP victory is hence 22,746. In this case we can see, since the LAB/CLP is a strong alternate coalition, that all the MOVC calculations agree with the MOV calculations. Note that this is not true for all seats. For example in the NSW data Sydney is the first seat where the MOVC (= 5583) for the LAB and CLP coalition is different from the MOV (=2864). This is because the runner-up was an Independent. Note that if we used MOV instead of MOVC we would incorrectly treat Sydney as one of the seats to change, and incorrectly calculate the number of votes required for an LAB/CLP coalition to win.

The full results for all seats are in Appendix A. The total numbers for changing the parliamentary outcome are computed by simply adding together the smallest margins for the necessary number of seats.

5 Conclusion

We have shown an efficient method of automated margin computation that can be used to identify the minimum number of vote changes (or errors) necessary to alter a parliamentary election outcome using single-member preferential voting. Our example was the NSW Legislative Assembly election of 2015, but the same tools and techniques could be immediately applied to any other parliament constructed in the same way for which full voting data was available, such as the Australian House of Representatives or other state lower houses.

Accurate electoral margins can form the basis of rigorous statistical auditing of paper ballot records to check the official election result. This would be valuable in any scenario, but is particularly important when an electronic (and hence unobservable) process such as automated ballot scanning is part of the count. Since these are exactly the scenarios that tend to produce detailed vote data, this work provides the basis for a count that is automated and fast (because of automated ballot scanning) and also transparent and verifiably accurate, because of rigorous auditing given an accurately computed election margin.

A Full list of margins for the NSW 2015 state election

Table 5 records the last-round and true victory margins for each seat in the 2015 NSW lower house elections. In most seats, the last-round margin – the difference between the two last candidates in the elimination order – is the true margin. Exceptions to this rule are marked with an asterisk. The 8 Liberal/National coalition seats with the smallest margins are shown in bold. The total of the margins of these 8 seats gives the smallest number of vote changes required to produce a hung parliament, 10,398.

Table 5: LRM and MOV for each seat in the 2015 NSW lower house election.

Seat	$\|\mathcal{C}\|$	LRM	MOV	Winner	Seat	$\|\mathcal{C}\|$	LRM	MOV	Winner
Gosford	6	102	102	LAB	The Entrance	5	171	171	LAB
East Hills	5	189	**189**	**LIB**	***Lismore**	6	1173	**209**	**NAT**
Strathfield	5	770	770	LAB	Granville	6	837	837	LAB
Upper Hunter	6	866	**866**	**NAT**	**Monaro**	5	1122	**1122**	**NAT**
*Balina	7	1267	1130	GRN	**Coogee**	5	1243	**1243**	**LIB**
Tweed	5	1291	**1291**	**NAT**	Prospect	5	1458	1458	LAB
Balmain	7	1731	1731	GRN	Rockdale	6	2004	2004	LAB
Port Stephens	5	2088	2088	CLP	Auburn	6	2265	2265	LAB
Penrith	8	2576	**2576**	**LIB**	Kogarah	6	2782	2782	LAB
Sydney	8	2864	2864	IND	**Holsworthy**	6	2902	**2902**	**LIB**
Goulburn	6	2945	2945	LIB	Oatley	5	3006	3006	LIB
Campbelltown	5	3096	3096	LAB	Newcastle	7	3132	3132	LAB
Wollongong	7	3367	3367	LAB	Macquarie Fields	7	3519	3519	LAB
Newtown	7	3536	3536	GRN	Heathcote	6	3560	3560	LIB
Blue Mountains	6	3614	3614	LAB	Myall Lakes	6	3627	3627	NAT
Bega	5	3663	3663	LIB	Wyong	7	3720	3720	LAB
Londonderry	5	3736	3736	LAB	Seven Hills	7	3774	3774	LIB
Summer Hill	7	3854	3854	LAB	Kiama	5	3856	3856	LIB
*Maitland	6	5446	4012	CLP	Terrigal	5	4053	4053	LIB
South Coast	5	4054	4054	LIB	Clarence	8	4069	4069	NAT
Lake Macquarie	7	4253	4253	IND	Mulgoa	5	4336	4336	LIB
Oxley	5	4591	4591	NAT	Tamworth	7	4643	4643	NAT
Maroubra	5	4717	4717	LAB	Swansea	8	4974	4974	LAB
Ryde	5	5153	5153	LIB	Barwon	6	5229	5229	NAT
Riverstone	5	5324	5324	LIB	Wagga Wagga	6	5475	5475	LIB

Continued

Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner
Parramatta	7	5509	5509	LIB	Charlestown	7	5532	5532	LAB
Bankstown	6	5542	5542	LAB	Blacktown	5	5565	5565	LAB
Coffs Harbour	5	5824	5824	NAT	*Heffron	5	5835	5824	LAB
Albury	5	5840	5840	LIB	Miranda	6	5881	5881	LIB
Mount Druitt	5	6343	6343	LAB	Canterbury	5	6610	6610	LAB
Fairfield	5	6998	6998	LAB	Epping	6	7156	7156	LIB
Bathurst	5	7267	7267	NAT	Hawkesbury	8	7311	7311	LIB
Wollondilly	6	7401	7401	LIB	Shellharbour	7	7519	7519	LAB
Cabramatta	5	7613	7613	LAB	Lane Cove	6	7740	7740	LIB
Drummoyne	6	8099	8099	LIB	Keira	5	8164	8164	LAB
Camden	5	8217	8217	LIB	Lakemba	5	8235	8235	LAB
Liverpool	5	8495	8495	LAB	North Shore	7	8517	8517	NAT
Murray	8	8574	8574	NAT	Hornsby	6	8577	8577	LIB
Dubbo	7	8680	8680	NAT	Port Macquarie	5	8715	8715	NAT
Cessnock	5	9187	9187	CLP	Cootamundra	5	9247	9247	NAT
Wallsend	5	9418	9418	LAB	Cronulla	5	9674	9674	LIB
Vaucluse	5	9783	9783	LIB	Baulkham Hills	5	10023	10023	LIB
Orange	5	10048	10048	NAT	Ku-ring-gai	5	10061	10061	LIB
*Willoughby	6	10247	10160	LIB	Wakehurst	6	10770	10770	LIB
Manly	5	10806	10806	LIB	Pittwater	5	11430	11430	LIB
Northern Tablelands	6	11969	11969	LIB	Davidson	5	12960	12960	LIB
Castle Hill	5	13160	13160	LIB					

Table 6 lists the number of vote changes (denoted $\varDelta$) necessary to elect an LAB or CLP candidate. This is at least the true margin (from the previous table), but may be strictly more, for example if an independent candidate was the runner-up. The rows inside the double lines are the 10 seats with the smallest changes necessary to give the labor parties 47 seats. The combined total number of votes needed to produce this is the sum of those rows: 22746.

Table 6: LRM, MOV, and the number of vote changes ($\varDelta$) required to elect an LAB or CLP candidate for each seat in the 2015 NSW lower house election.

Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	$\varDelta$	Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	$\varDelta$
Auburn	6	2265	2265	LAB	0	Bankstown	6	5542	5542	LAB	0
Blacktown	5	5565	5565	LAB	0	Blue Mountains	6	3614	3614	LAB	0
Cabramatta	5	7613	7613	LAB	0	Campbelltown	5	3096	3096	LAB	0
Canterbury	5	6610	6610	LAB	0	Cessnock	5	9187	9187	CLP	0
Charlestown	7	5532	5532	LAB	0	Fairfield	5	6998	6998	LAB	0
Gosford	6	102	102	LAB	0	Granville	6	837	837	LAB	0
Heffron	5	5835	5824	LAB	0	Keira	5	8164	8164	LAB	0
Kogarah	6	2782	2782	LAB	0	Lakemba	5	8235	8235	LAB	0
Liverpool	5	8495	8495	LAB	0	L-derry	5	3736	3736	LAB	0
Macq. Fields	7	3519	3519	LAB	0	Maitland	6	5446	4012	CLP	0
Maroubra	5	4717	4717	LAB	0	Mt. Druitt	5	6343	6343	LAB	0

Continued

Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	$\varDelta$	Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	$\varDelta$
Newcastle	7	3132	3132	LAB	0	P. Stephens	5	2088	2088	CLP	0
Prospect	5	1458	1458	LAB	0	Rockdale	6	2004	2004	LAB	0
Shellharbour	7	7519	7519	LAB	0	Strathfield	5	770	770	LAB	0
Summer Hill	7	3854	3854	LAB	0	Swansea	8	4974	4974	LAB	0
The Entrance	5	171	171	LAB	0	Wallsend	5	9418	9418	LAB	0
Wollongong	7	3367	3367	LAB	0	Wyong	7	3720	3720	LAB	0
East Hills	5	189	189	LIB	189	Lismore	6	1173	209	NAT	209
U. Hunter	6	866	866	NAT	866	Monaro	5	1122	1122	NAT	1122
Balina	7	1267	1130	GRN	1130	Coogee	5	1243	1243	LIB	1243
Tweed	5	1291	1291	NAT	1291	Balmain	7	1731	1731	GRN	1731
Penrith	8	2576	2576	LIB	2576	Holsworthy	6	2902	2902	LIB	2902
Goulburn	6	2945	2945	LIB	2945	Oatley	5	3006	3006	LIB	3006
Newtown	7	3536	3536	GRN	3536						
Heathcote	6	3560	3560	LIB	3560	M. Lakes	6	3627	3627	NAT	3627
Bega	5	3663	3663	LIB	3663	Seven Hills	7	3774	3774	LIB	3774
Kiama	5	3856	3856	LIB	3856	Terrigal	5	4053	4053	LIB	4053
South Coast	5	4054	4054	LIB	4054	Clarence	8	4069	4069	NAT	4069
Lake Macq.	7	4253	4253	IND	4253	Mulgoa	5	4336	4336	LIB	4336
Oxley	5	4591	4591	NAT	4591	Ryde	5	5153	5153	LIB	5153
Barwon	6	5229	5229	NAT	5229	Riverstone	5	5324	5324	LIB	5324
W-Wagga	6	5475	5475	LIB	5475	Parramatta	7	5509	5509	LIB	5509
Sydney	8	2864	2864	IND	5583	C. Harbour	5	5824	5824	NAT	5824
Albury	5	5840	5840	LIB	5840	Miranda	6	5881	5881	LIB	5881
Epping	6	7156	7156	LIB	7156	Bathurst	5	7267	7267	NAT	7267
Hawkesbury	8	7311	7311	LIB	7311	W-dilly	6	7401	7401	LIB	7401
Lane Cove	6	7740	7740	LIB	7740	D-moyne	6	8099	8099	LIB	8099
Camden	5	8217	8217	LIB	8217	Hornsby	6	8577	8577	LIB	8577
Dubbo	7	8680	8680	NAT	8680	Port Macq.	5	8715	8715	NAT	8715
North Shore	7	8517	8517	NAT	8798	C-mundra	5	9247	9247	NAT	9247
Murray	8	8574	8574	NAT	9483	Cronulla	5	9674	9674	LIB	9674
B. Hills	5	10023	10023	LIB	10023	Orange	5	10048	10048	NAT	10048
Ku-ring-gai	5	10061	10061	LIB	10061	Willoughby	6	10247	10160	LIB	10160
Vaucluse	5	9783	9783	LIB	10581	Wakehurst	6	10770	10770	LIB	10770
Tamworth	7	4643	4643	NAT	11283	N. T-lands	6	11969	11969	LIB	11969
Manly	5	10806	10806	LIB	12106	Pittwater	5	11430	11430	LIB	12181
Davidson	5	12960	12960	LIB	13065	Castle Hill	5	13160	13160	LIB	13160

Table 7 records the margins for a Labor-Green coalition. In this case the total number of vote changes required to produce this outcome is 16349.

Table 7: LRM, MOV, and the number of vote changes ($\varDelta$) required to elect a LAB, CLP, or GRN for each seat in the 2015 NSW lower house election.

Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	$\varDelta$	Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	$\varDelta$
Auburn	6	2265	2265	LAB	0	Balina	7	1267	1130	GRN	0

Continued

Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	Δ	Seat	$\lvert\mathcal{C}\rvert$	LRM	MOV	Winner	Δ
Balmain	7	1731	1731	GRN	0	Bankstown	6	5542	5542	LAB	0
Blacktown	5	5565	5565	LAB	0	B. Mountains	6	3614	3614	LAB	0
Cabramatta	5	7613	7613	LAB	0	C-belltown	5	3096	3096	LAB	0
Canterbury	5	6610	6610	LAB	0	Cessnock	5	9187	9187	CLP	0
Charlestown	7	5532	5532	LAB	0	Fairfield	5	6998	6998	LAB	0
Gosford	6	102	102	LAB	0	Granville	6	837	837	LAB	0
Heffron	5	5835	5824	LAB	0	Keira	5	8164	8164	LAB	0
Kogarah	6	2782	2782	LAB	0	Lakemba	5	8235	8235	LAB	0
Liverpool	5	8495	8495	LAB	0	L-derry	5	3736	3736	LAB	0
M. Fields	7	3519	3519	LAB	0	Maitland	6	5446	4012	CLP	0
Maroubra	5	4717	4717	LAB	0	Mt. Druitt	5	6343	6343	LAB	0
Newcastle	7	3132	3132	LAB	0	Newtown	7	3536	3536	GRN	0
P. Stephens	5	2088	2088	CLP	0	Prospect	5	1458	1458	LAB	0
Rockdale	6	2004	2004	LAB	0	S-harbour	7	7519	7519	LAB	0
Strathfield	5	770	770	LAB	0	S. Hill	7	3854	3854	LAB	0
Swansea	8	4974	4974	LAB	0	The Entr.	5	171	171	LAB	0
Wallsend	5	9418	9418	LAB	0	Wollongong	7	3367	3367	LAB	0
Wyong	7	3720	3720	LAB	0						
East Hills	5	189	189	LIB	189	Lismore	6	1173	209	NAT	209
U. Hunter	6	866	866	NAT	866	Monaro	5	1122	1122	NAT	1122
Coogee	5	1243	1243	LIB	1243	Tweed	5	1291	1291	NAT	1291
Penrith	8	2576	2576	LIB	2576	Holsworthy	6	2902	2902	LIB	2902
Goulburn	6	2945	2945	LIB	2945	Oatley	5	3006	3006	LIB	3006
Heathcote	6	3560	3560	LIB	3560	M. Lakes	6	3627	3627	NAT	3627
Bega	5	3663	3663	LIB	3663	Seven Hills	7	3774	3774	LIB	3774
Kiama	5	3856	3856	LIB	3856	Terrigal	5	4053	4053	LIB	4053
S. Coast	5	4054	4054	LIB	4054	Clarence	8	4069	4069	NAT	4069
Lake Macq.	7	4253	4253	IND	4253	Mulgoa	5	4336	4336	LIB	4336
Oxley	5	4591	4591	NAT	4591	Ryde	5	5153	5153	LIB	5153
Barwon	6	5229	5229	NAT	5229	Riverstone	5	5324	5324	LIB	5324
W-Wagga	6	5475	5475	LIB	5475	Parramatta	7	5509	5509	LIB	5509
Sydney	8	2864	2864	IND	5583	C. Harbour	5	5824	5824	NAT	5824
Albury	5	5840	5840	LIB	5840	Miranda	6	5881	5881	LIB	5881
Epping	6	7156	7156	LIB	7156	Bathurst	5	7267	7267	NAT	7267
H-bury	8	7311	7311	LIB	7311	W-dilly	6	7401	7401	LIB	7401
Lane Cove	6	7740	7740	LIB	7740	D-moyne	6	8099	8099	LIB	8099
Camden	5	8217	8217	LIB	8217	N. Shore	7	8517	8517	NAT	8517
Hornsby	6	8577	8577	LIB	8577	Dubbo	7	8680	8680	NAT	8680
Port Macq.	5	8715	8715	NAT	8715	C-mundra	5	9247	9247	NAT	9247
Murray	8	8574	8574	NAT	9483	Cronulla	5	9674	9674	LIB	9674
Vaucluse	5	9783	9783	LIB	9783	B. Hills	5	10023	10023	LIB	10023
Orange	5	10048	10048	NAT	10048	Ku-ring-gai	5	10061	10061	LIB	10061
Willoughby	6	10247	10160	LIB	10160	Wakehurst	6	10770	10770	LIB	10770

Continued

```
margin-irv($\mathcal{C}$, $\mathcal{B}$, $c_w$, $\mathcal{A}$)
1  $F := \emptyset$
2  $U := LRM_{\mathcal{B}}$
3  for($c \in \mathcal{A}$)
4      $\pi' := [c]$
5      $l :=$ LowerBound($\pi'$)
6      if($l < U$)
7          $F := F \cup \{(l, \pi')\}$
8  while $F \neq \emptyset$
9      $(l, \pi') := \arg\min F$
10     $F := F \setminus \{(l, \pi')\}$
11     $U :=$ expand($l, \pi', U, F, \mathcal{C}, \mathcal{B}$)
12 return $U$

expand($l, \pi', U, F, \mathcal{C}, \mathcal{B}$)
13 $l' := \max\{l,$ DistanceTo($\pi'$, $\mathcal{C}$,$\mathcal{B}$)$\}$
14 if($l' \geq U$)
15     return $U$
16 for($c \in \mathcal{C} \setminus \pi'$)
17     $\pi := [c] +\!\!+ \pi'$
18     if($|\pi| = |\mathcal{C}|$)
19         return $\min\{U,$DistanceTo($\pi$,$\mathcal{C}$,$\mathcal{B}$)$\}$
20     $l'' = \max\{l',$ LowerBound($\pi$)$\}$
21     if($l'' < U$)
22         $F := F \cup \{(l'', \pi)\}$
23 return $U$
```

Fig. 2: MOVC computation for an IRV election $\mathcal{B}$ with candidates $\mathcal{C}$, winner $c_w \in \mathcal{C}$, and alternate winner set $\mathcal{A}$.

Seat	$\|\mathcal{C}\|$	LRM	MOV	Winner	Δ	Seat	$\|\mathcal{C}\|$	LRM	MOV	Winner	Δ
Manly	5	10806	10806	LIB	10806	Tamworth	7	4643	4643	NAT	11283
Pittwater	5	11430	11430	LIB	11430	N. T-lands	6	11969	11969	LIB	11969
Davidson	5	12960	12960	LIB	12960	Castle Hill	5	13160	13160	LIB	13160

B Modified *margin-irv*: Computing the MOVC

The *margin-irv* algorithm for computing the MOVC for an IRV election $\mathcal{B}$ given a set of alternate winners $\mathcal{A}$ is shown in Figure 2. An initial upper bound on the MOVC is initialised to the last round margin ($LRM_{\mathcal{B}}$) in Step 2. For each candidate in $\mathcal{A}$, we add a partial elimination order to our frontier F. Each order π' is assigned a lower bound (computed as described by Blom *et al.* [1]) on the degree of manipulation required to realise an elimination sequence *ending* in π' – only orders with an estimated lower bound (l) that *is less than* the current MOVC upper bound (U) are added to the frontier (Steps 6 and 7). Steps 8 to 12 repeatedly select the partial order π' in F with the smallest associated lower bound for expansion. To expand an order π', we create a new order for each candidate c *not* already present in π', appending c to the start of the sequence (Step 17). If the created sequence π contains all candidates, it is a leaf node, and we evaluate the exact number of vote changes required to realise the sequence with a mixed integer linear program (MIP) denoted DistanceTo.

Section B.1 provides the formulation of the DistanceTo MIP, replicated from Blom *et al.* [1]. Otherwise, we compute a lower bound on the on the degree of manipulation required to realise an elimination sequence *ending* in π (l'') and add π to our frontier if this lower bound is less than our current upper bound on the MOVC (Steps 21 to 22). The algorithm terminates once there are no further partial orders to be expanded in our frontier, returning the current MOVC upper bound (U) as the computed MOVC.

B.1 The DISTANCETO MIP

The following MIP formulation, originally presented in the work of Magrino *et al.* [5], has been replicated as it appears in Blom *et al.* [1]. Let $\mathbf{R}$ denote the set of possible (partial and total) rankings R of candidates $\mathcal{C}$ that could appear on a vote, N_R the number of votes cast in the election with ranking $R \in \mathbf{R}$, and N the total number of votes cast. For each $R \in \mathbf{R}$, we define variables:

q_R integer number of votes to be changed into R;
m_R integer number of votes with ranking R in the unmodified election to be changed into something other than R; and
y_R number of votes in the modified election with ranking R.

Given a partial or complete order π, the DISTANCETO MIP is:

$$\min \sum_{R \in \mathbf{R}} q_R$$

$$N_R + q_R - m_R = y_R \qquad \forall R \in \mathbf{R} \quad (3)$$

$$\sum_{R \in \mathbf{R}} q_R = \sum_{R \in \mathbf{R}} m_R \qquad (4)$$

$$\sum_{R \in \mathcal{R}_{i,i}} y_R \leq \sum_{R \in \mathcal{R}_{j,i}} y_R \qquad \forall c_i, c_j \in \pi \ . \ i < j \quad (5)$$

$$n \geq y_R \geq 0, \ \ N_R \geq m_R \geq 0, \ \ q_R \geq 0 \qquad \forall R \in \mathbf{R} \quad (6)$$

Constraint (3) states that the number of votes with ranking $R \in \mathbf{R}$ in the new election is equal to the sum of those with this ranking in the unmodified election and those whose ranking has *changed to* R, minus the number of votes whose ranking has been *changed from* R. Constraint (5) defines a set of *special elimination constraints* which force the candidates in π to be eliminated in the stated order. $\mathcal{R}_{j,i}$ denotes the subset of rankings in $\mathbf{R}$ ($\mathcal{R}_{j,i} \subset \mathbf{R}$) in which c_j is the most preferred candidate still standing (i.e., that will count toward c_j's tally) at the start of round i (in which candidate c_i is eliminated). Constraint (4) ensures that the total number of votes cast in the election does not change as a result of the manipulation.

References

1. M. Blom, P. J. Stuckey, V. Teague, and R. Tidhar. Efficient Computation of Exact IRV Margins. In *European Conference on AI (ECAI)*, pages 480–487, 2016.
2. S. Jackman. Measuring electoral bias: Australia, 1949–93. *British Journal of Political Science*, 24(3):319–357, 1994.
3. J. A. Kroll, J. A. Halderman, and E. W. Felten. Efficiently auditing multi-level elections. *Ann Arbor*, 1001:48109. https://jhalderm.com/pub/papers/audit-evote14.pdf.
4. M. Lindeman and P. B. Stark. A gentle introduction to risk-limiting audits. *IEEE Security & Privacy*, 10(5):42–49, 2012.

5. T. R. Magrino, R. L. Rivest, E. Shen, and D. A. Wagner. Computing the margin of victory in IRV elections. In *USENIX Accurate Electronic Voting Technology Workshop*, USENIX Association Berkeley, CA, USA, 2011.
6. Parliament of Australia Joint Standing Committee on Electoral Matters. Third interim report on the inquiry into the conduct of the 2016 federal election: AEC modernisation, June 2017. http://www.aph.gov.au/Parliamentary_Business/ Committees/Joint/Electoral_ Matters/2016Election/Third_Interim_Report.
7. E. R. Tufte. The relationship between seats and votes in two-party systems. *American Political Science Review*, 67(2):540–554, 1973.
8. W. C. Yang. Democracy, minimized-given various election scenarios, what is the minimum percent of the popular vote required to win the white house? *OR MS Today*, 35(5):34, 2008.

Attacks

The Threat of SSL/TLS Stripping to Online Voting

Anthony Cardillo and Aleksander Essex

Department of Electrical and Computer Engineering
Western University, London, ON, Canada
`{acardill,aessex}@uwo.ca`

Abstract. In many real-world deployments of online voting, Transport Layer Security (TLS) represents the primary (and in some cases *only*) line of defense against network based man-in-the-middle attacks that can steal voter credentials and modify ballot selections. In this paper we examine online voting in the context of *TLS stripping attacks*, which exploit the situation where a voter types or clicks a URL of the form `example.com` or `http://example.com`. Despite the widespread availability of effective protections, we present a study of voting-related websites finding the overwhelming majority are vulnerable to TLS stripping to some degree, with most offering no explicit protection at all.

1 Introduction

Recall the last time you logged in to your social network account, an online retailer, financial institution, or even online election. You should have seen a lock icon in the address bar of your browser. Was it there? Did you check? Suppose the icon was missing. Would you notice? And if you did notice, what would you attribute it's absence to? Maybe you misunderstood the security indicators. Maybe the server was mis-configured. Or maybe you were the victim of a cyber attack that prevented your browser from initiating a secure connection using transport-layer security (TLS), enabling attackers to monitor and/or modify the content you send and receive.

TLS stripping[1] [15, 4] is a network based man-in-the-middle attack which suppresses or *strips* TLS from a communication channel. The attack is made possible when a user types or clicks a non-HTTPS URL of the form `example.com` or `http://example.com` (as opposed to `https://example.com`). This instructs the browser to make an insecure request over HTTP instead of its encrypted and authenticated counterpart, HTTPS. A well configured TLS-enabled server would typically respond to such a request by directing the client to request the resource over HTTPS instead. A man-in-the-middle can intercept and suppress this response, and continue communicating with the client over HTTP. Since the client requested an HTTP connection to begin with, the missing TLS redirect

[1] More commonly known as *SSL Stripping*, it was originally named after the now-deprecated Secure Sockets Layer (SSL) protocol.

goes unnoticed, and the man-in-the-middle is free to observe and modify any data exchanged in the interaction.

Although TLS stripping is a significant and effective threat to online security generally, it has only briefly been considered in the context of elections [25, 10]. In this paper we conducted a study of the adoption of TLS stripping mitigations by election websites. We found election websites systematically lagging in the adoption of industry best-practices. We examined an international cross-section of over 100 election, vendor, and voter registration websites and found 98% were vulnerable to TLS stripping to some degree, with 84% providing no mitigation at all. We also found a number of servers with serious TLS vulnerabilities, which we disclosed to the affected organizations.

2 Motivation

Online voting is a unique use case of the web, with a confluence of factors that increase the severity of TLS stripping attacks. The factors that warrant further study of this topic are as follows.

Critical Infrastructure. Online voting websites must conform to higher cybersecurity standards. Elections are increasingly being recognized as critical infrastructure. In 2017, for example, the U.S. Department of Homeland Security designated elections systems as critical infrastructure under the Government Facilities Sector.[2] As such, it is important to understand how online voting websites are performing relative to industry security standards.

Secret Ballot, Secret Tampering. As we discuss in Section 3.1, TLS represents the main (and in some cases *only*) line of defense against network based man-in-the-middle attacks. Unlike other online settings (e.g., social media, online banking, etc.), an attack stealing voter credentials or modifying ballot selections can be more difficult to detect and correct due to ballot secrecy requirements, making the impact of a man-in-the-middle attack more severe in comparison.

Communicating URLs to Voters. The transient nature of elections often makes communicating the URL of an election website to voters a weak link in a literal and figurative sense. As discussed in Section 3, we found numerous cases of election officials and candidates explicitly directing voters to use HTTP URLs.

Systematic Lack of Best Practice Adoption. Effective mitigations for TLS stripping have long been adopted by leading websites like Google, Amazon, Facebook, etc. If elections truly are to serve as critical infrastructure, they must provide a degree of web security that is *no worse* than current industry practices. As our findings in Section 4 show, however, almost every election-related website we examined is vulnerable to TLS stripping.

[2] `https://www.dhs.gov/government-facilities-sector`

Barriers to Adoption. Unlike other vulnerabilities that can be resolved with software updates, mitigations to TLS stripping are opt-in, meaning the organization has to be aware of the attack, be aware of the mitigations, and take action to apply them. As we discuss in Section 5, election agencies and vendors face a variety of obstacles in this regard.

3 The Threat of TLS Stripping to Online Elections

Transport Layer Security (TLS) is a standardized group of cryptographic protocols for secure network communication [18, 19], providing confidentiality, integrity and authentication of network communications at the application layer.

Attacks against TLS typically involve directly attacking components, typically by exploiting vulnerabilities in the software implementation, the cryptographic primitives, or the protocol itself. Recent examples include key-recovery attacks on export strength ciphersuites [23, 1], buffer over-read vulnerabilities [8], insufficient public-key validation [7, 22], and eavesdropping attacks exploiting either TLS compression (BEAST and CRIME) and padding oracles [11]. We refer the reader to the surveys of Clark and van Oorschot [4] and Sheffer et al. [20] for a systematic study of known attacks.

TLS stripping differs from these approaches. It does not exploit a concrete TLS vulnerability, but rather prevents a TLS connection from being established in the first place.

3.1 Role of TLS in Online Elections

Suppose a voter types the URL of an election website, `voting-site.com`. The server responds with the login page, and the voter enters their user id and password. The browser sends an HTTP `POST` over TLS to the login page, e.g., `https://voting-site.com/login.php` with the following contents:

```
auth_id=1234&auth_pwd=123456&submit=Login.
```

Most login pages do not use encryption or authentication inside of the TLS connection. Without it, or with a vulnerable implementation, a man-in-the-middle can directly recover the voter's credentials from the contents of the login `POST`. Now suppose the voter marks a vote for a candidate, Alice, and clicks on the *Cast* button. The browser makes an HTTP `POST` to the cast ballot page `https://voting-site.com/cast.php` with the contents:

```
president=Alice&election_id=US_2020&submit=Confirm
```

Since no other encryption or authentication exists on the *POST* contents, a man-in-the-middle can arbitrarily change the voter's selections, e.g., by setting `president=Bob`, or perhaps more simply by swapping candidate names in the ballot HTML.

Application Layer Encryption. Some online voting designers have attempted to mitigate the consequence of TLS based attacks by employing additional cryptographic protections at the application layer. For example, the iVote system in Australia[3] uses client-side Javascript to encrypt `POST` data. The problem remains, however, that the Javascript is delivered to the client over TLS. Teague and Halderman [12] demonstrated the ability to inject vote-stealing Javascript as a result of the use of weak crypographic parameters [23] in the state election of New South Wales in 2014. Even without actively injecting Javascript, Culnane et al. [5] demonstrated the feasibility of passively recovering voter credentials in iVote using brute force methods.

Multi-factor Authentication. Other online voting designers mitigate TLS attacks by employing additional voter authentication factors. Online voting in the Estonian system [21], for example, is done through a special election-specific software application. The approach removes much of the user from the equation by forcing TLS and using a pre-loaded (pinned) certificate to authenticate the election server. Additionally, the Estonian system uses national Electronic ID cards to digitally sign ballots (although Nemec et al. [17] recently demonstrated a major signature forgery vulnerability in the Estonian PKI). While these additional factors add defense in depth with regard to TLS stripping attacks, both the voting application[4] and the electronic ID software[5] are still initially downloaded by the voter in a web browser over TLS.

3.2 TLS Stripping

TLS stripping (originally *SSL* stripping) was introduced Moxie Marlinspike [15] on the observation that most TLS connections occurred only when a user either clicked on an explicit HTTPS link, or when the server sent an HTTPS redirect. As he noted: “Nobody types ‘https://’, or even ‘http://’ for that matter.” And when a user either types or clicks on a URL of the form `example.com` (or `http://example.com`), a TLS-enabled server typically responds by directing the client to request the resource securely over HTTPS instead. The client would then initiate a TLS handshake, after which time the interaction would continue over an encrypted and authenticated connection (See Figure 1). A man-in-the-middle can intercept and suppress this redirect, and continue the interaction with the client over HTTP. The man-in-the-middle can then initiate and maintain its own TLS connection with the server.

Because the client requested an HTTP connection to begin with, the missing TLS redirect, and continued interaction over the insecure channel goes unnoticed, and the man-in-the-middle is free to observe or modify any messages between the client and server (See Figure 2).

[3] `https://ivote.nsw.gov.au`
[4] `https://valimised.ee`
[5] `https://www.id.ee`

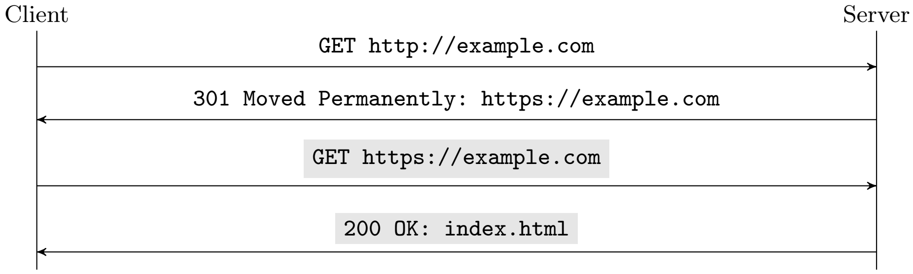


Fig. 1. Normal HTTPS Redirect. When a user types or clicks an HTTP URL, a well configured TLS-enabled server would direct the client to use HTTPS instead, and the interaction would continue over a secure TLS connection (shown in grey).

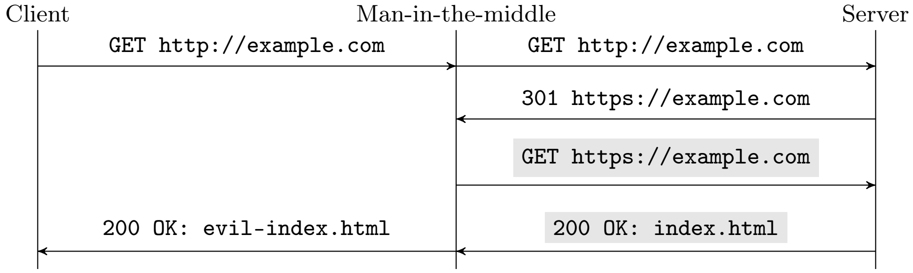


Fig. 2. TLS Stripping Attack. A man-in-the-middle can intercept a request made by a client over HTTP and proxy it to the server via its own separate TLS connection (shown in grey). Since the client never saw the HTTPS redirect, the man-in-the-middle is free to continue the connection over HTTP.

Detecting TLS Stripping. TLS stripping attacks cannot not be reliably detected by users in practice, as they do not generate browser errors or warnings. Instead, they require the user to notice the *absence* of security indicators such as the `https://` in the URL, and the padlock icon in the address bar [6, 2]. Inconsistent and confusing security indicators add to the challenge [3]. Some browsers such as Safari and Edge do not even display the `HTTPS` indicator, and employ a subtle padlock icon (see Figure 3). Due to limited screen resolution, many mobile browsers hide the address bar when scrolling. Even if a user consciously registers the unexpected behavior, research has suggested that voters are not likely attribute it to a malicious cause [16].

3.3 HTTP Strict Transport Security (HSTS)

When TLS stripping was first introduced, there was no mechanism to definitively declare preference for HTTPS, which led to the development of the HTTP Strict Transport Security (HSTS) standard [13]. An HSTS directive can be placed in

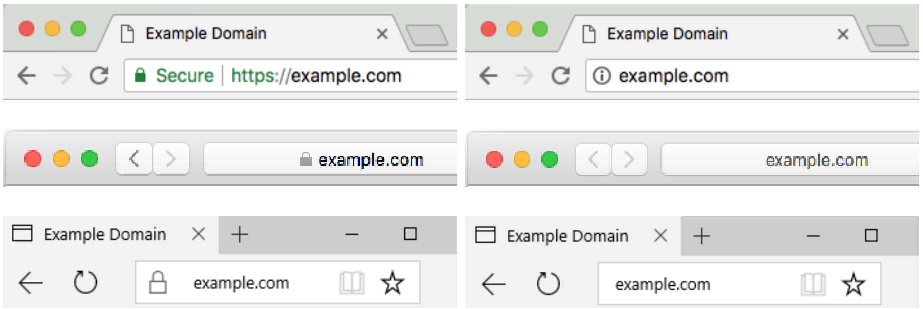


Fig. 3. Attack indicators Across Browsers. Secure connections (left) versus TLS stripped connections (right) in the desktop versions of Chrome (top), Safari (middle), and Edge (bottom).

an HTTP response header allowing a server to advertise its intention to only communicate over HTTPS for all future connections. This includes an expiry period (expressed in seconds), and optional fields to include subdomains, and express consent to be added to the preload list. For example, an HTTP response header containing

```
Strict-Transport-Security: max-age=31536000; includeSubDomains;
                           preload;
```

would direct the browser to store the preference in cache for 1 year, for all subdomains, and would direct browser developers to add the site to the preload list. From this point forward until expiry (or until the user clears the browser's cache), the browser will make all requests to that domain (and any subdomains) over HTTPS—even if the user types or clicks an HTTP link.

Adoption of the HSTS standard has been slow but steady. At the time of writing, 16.8% of the top 150,000 TLS-enabled sites supported HSTS.[6] As of 2015, the US government instituted a policy requiring all executive (i.e., .gov) departments and agencies to use HTTPS and enable HSTS.[7]

3.4 HSTS Preload list

HSTS uses a trust-on-first-use model. To be effective in preventing a TLS stripping attack, a user must have previously visited the website in order to have received and stored the server's preference. There are, however, a number of situations where this requirement would not be met, such as when: the user has never visited the site before; the browser, operating system or device was recently upgraded or switched; the user recently cleared their browser's cache; or, the user had not visited the site for an amount of time exceeding the max-age.

[6] https://www.ssllabs.com/ssl-pulse

[7] US Office of Management and Budget. Memorandum M-15-13, 2015. https://https.cio.gov

For many websites, and especially election websites, it may not be reasonable to assume a previous visit has recently occurred. In addition, some URLs may never be visited. For example, a visit to `example.com` might redirect to `https://www.example.com`. Since the client never visited `https://example.com`, it will not receive an `includeSubDomains` directive that applies to the entire zone, and any subdomains (e.g., `sub.example.com`) will be left unprotected.

For these reasons, the Chromium security team created a list[8] of HSTS-enabled domains *preloaded* into Chrome, and each user receives this list automatically whenever Chrome is installed or updated. Visiting any domain on this list is done over HTTPS only (even on a first visit), meaning no HTTP to HTTPS redirects are ever made, and therefore no opportunity for TLS stripping exists. Other browsers like Firefox, Opera, Safari and IE/Edge use preload lists based on the Chrome list.

In 2014 the list contained only 233 non-Google domains [14], and additions were handled manually over email. At the time of writing there were 50,000 domains on the list, spanning a wide variety of sites ranging from large companies and government agencies, to small businesses and personal websites, and addition is handled by an automated submission site.[9] Although the `.gov` domains account for less then 1% of the preload list, the DotGov registrar has begun automatically preloading all newly registered `.gov` domains.

3.5 Communicating Voting Website URLs

One of the major challenges of hosting an online election is communicating authentic URLs to voters. In an effort to increase voter turnout, election officials and campaigns may employ a variety of modes of communication. A vulnerability arises, however, if the voting website does not employ HSTS preloading *and* the user types or clicks on an HTTP URL. As we will see in the following section, this first circumstance occurs in the vast majority of election websites. But how likely is a user to initially visit the voting site over HTTP? User testing is beyond the scope of this study, however we observed numerous situations in which the voter was explicitly directed to the voting site over HTTP (see Figure 4).

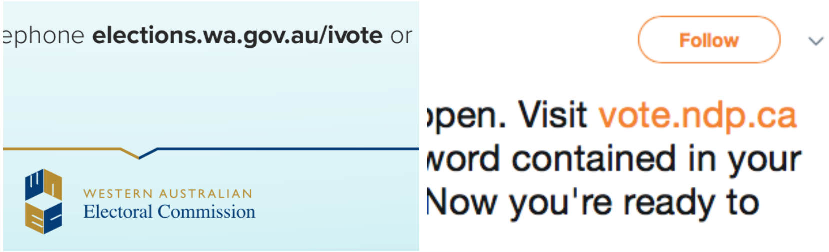
For example, during the New Democratic Party of Canada's leadership election in October 2017, we observed numerous HTTP links on social media originating from the accounts of candidates, riding associations, supporters, and even the party itself. In the mailer to voters, the instructions explicitly directed voters to the website via HTTP: "How can I vote online? Type vote.ndp.ca into the address bar of your web browser."

In another example, the Ontario Labour Relations Board held an online strike vote for college teachers in November 2017. Not only was the landing page to the voting site emailed as an HTTP link to voters (`olrb-crto.isivote.com`), the site was not available over HTTPS. Only when voters selected their preferred

[8] https://www.chromium.org/hsts

[9] https://hstspreload.org

[11] https://twitter.com/NDP/status/910253791718645765



ephone **elections.wa.gov.au/ivote** or

WESTERN AUSTRALIAN
Electoral Commission

Follow

›pen. Visit vote.ndp.ca
word contained in your
Now you're ready to

(a) Western Australia Election Commission, Poster, 2017.

(b) New Democratic Party of Canada, Twitter, 2017

Fig. 4. The Weakest Link. Examples from recent online elections of voters being directed to type [24] or click[11] insecure HTTP URLs.

language were they redirected to a TLS enabled site (on a different domain) to log in.

4 Study of HSTS Preload Adoption in Election-related Websites

This section presents a study of HSTS pre-loading among websites with an election focus. We examined a number of websites, including the online voting sites of specific elections, government agencies responsible for election administration and voter registration, and the websites of companies offering online voting solutions. In total we examined 103 election sites, and chose to focus on Australia, Canada, Switzerland and the United States as countries that have a high concentration of websites pertaining to elections.

Each site was evaluated for the availability of a well-configured TLS configuration, the presence of HSTS headers with a non-trivial max-age, and membership in the HSTS preload list. We comment on any vulnerabilities in the TLS configuration for sites that scored a grade of C or lower on Qualsys SSL Test.[12]

4.1 Election Websites

Only a handful of online elections occur at any time, and many sites stay online only during the polling period. For this study, our goal was to present a diverse (not necessarily complete) sample of voting sites. We examined the TLS configuration of voting sites with active login pages during mid May 2018, and present our findings in Table 1, which revealed minimal protection against TLS stripping attacks.

Of the 10 sites examined, only two were on the preload list. The first is Helios, an end-to-end verifiable internet voting scheme (E2E-VIV) [9], which has been

[12] https://ssllabs.com/ssltest

Table 1. Snapshot of HSTS in Online Voting Websites

Domain	Election	TLS	HSTS	Preload
`ivote-cvs.elections.wa.gov.au`	Western Australia State 2017	●	○	○
`esc-vote.com/acm2018`	ACM General Election 2018	●	○	○
`evoting.ch`	Swiss Post E-Voting	●	●	●
`heliosvoting.org`	Helios	●	●	●
`intvoting.com/OntarioPC`	Ontario PC Leadership 2018	●	○	○
`innskraning.island.is`	Iceland Citizen E-referendum	●	○	○
`ivote.nsw.gov.au`	New South Wales State 2015	●	●	○
`olrb-crto.isivote.com`	Ontario Labour Board 2018	○	○	○
`valimised.ee`	Elections Estonia	●	○	○
`vote.ndp.ca`	NDP Canada Leadership 2017	●	○	○

used to conduct elections for organizations such as the International Association of Cryptologic Researchers (IACR) and the Princeton Graduate Students' Association. We contacted the Helios maintainers in October 2017 asking them to consider adding `heliosvoting.org` to the preload list. They did, and (to our knowledge) became the first online voting site on the preload list. The second is Swiss Post's E-Voting site, which has been used by the cantons of Fribourg and Neuchâtel, and will be used for the first time in upcoming elections in the cantons of Basel-Stadt, Glarus, and Thurgau. We observed inbound links to the E-Voting site originating from the canton sites. We examined these sites and found none were using HSTS or preloading, and some even did not offer TLS (see Table 2). Voters attempting to visit via `evoting.ch` via the canton websites would, therefore, still be vulnerable to TLS stripping.

Table 2. HSTS in Swiss Cantons Using Online Voting

Domain	Canton	TLS	HSTS	Preload
`bs.ch`	Basel-Stadt	○	○	○
`fr.ch`	Fribourg	●	○	○
`gl.ch`	Glarus	◐[a,b]	○	○
`ne.ch`	Neuchâtel	○	○	○
`tg.ch`	Thurgau	◐[c]	○	○

[a] Vulnerable to POODLE (`CVE-2014-3566`).
[b] Uses weak/obsolete ciphersuites.
[c] Vulnerable to ROBOT (`CVE-2017-13099` and others).

4.2 Online Voting Vendor Websites

We studied the corporate websites of vendors offering online voting solutions. Vendors are the implementors of online voting sites, and although their own sites are typically not directly associated with ballot casting, we would contend that they serve as an indicator of a vendor's awareness and ability to follow best practices. A selection of 27 different online voting vendors was studied (see Table 3).

At the time of inspection, 3 domains exhibited insufficient TLS protection; Dominion and Election Service Co. were improperly serving their cloud provider's certificate, which generated a browser error. Voting Place was offering SSL 3.0 which is vulnerable to `CVE-2014-3566`. Of the remaining domains, only 9 implemented HSTS, and no sites were found on the preload list.

4.3 Election Agencies

Critical election information such as dates and polling locations must be communicated to voters. While web-based attacks against election agencies providing such information would not impact the vote directly, it could serve to suppress votes, and undermine public trust. For example, an automated phone scam was used in Guelph, Canada in 2011 to direct voters to a fake polling location, and led to a criminal conviction,[13] which might be harder to achieve in the more anonymous setting of the web.

We decided to study the configurations of election agency websites in two countries who have used online voting at the sub-national level: Australia (see Table 4) and Canada (see Table 5). Of the 14 federal, provincial and territorial election agencies in Canada, none used HSTS or preloading, 2 had serious TLS configuration issues, and 9 did not implement TLS protection whatsoever. Elections British Columbia was found serving a self-signed certificate, which caused a browser error. We contacted them, and they promptly responded by obtaining and installing a certificate with a valid trust path. Élections Québec acknowledged receipt of our recommendations to disable weak and obsolete ciphersuites (esp. those using RC4), but had still not done so at time of writing.

Australia's election agencies fared slightly better: all 9 domains used TLS, however HSTS was found on a single domain only, and none were on the preload list.

4.4 Voter Registration Websites

Another important online election-related activity is voter registration. Some governments allow voters to sign up to vote and access information such as registration details, contact information of elected officials, voting instructions, location of ballot drop boxes and voting centers. For this component, we decided to focus on the Unites States. Most states in the US offer online voter registration.

[13] R. v. Sona, 2016 ONCA 452, Court of Appeals for Ontario, 2016. http://www.ontariocourts.ca/decisions/2016/2016ONCA0452.pdf

Table 3. HSTS in Online Voting Vendor Websites

Domain	Company	TLS	HSTS	Preload
agora.vote	Agora	●	○	○
coalichain.io	Coalichain	●	○	○
cyber.ee	Cybertentica	●	●	○
democracy.earth	Democracy Earth	●	○	○
dominionvoting.com	Dominion Voting	◐[a]	○	○
eballot.com	eBallot	●	○	○
electionrunner.com	Election Runner	●	○	○
electionservicesco.com	Election Services Co.	◐[a]	○	○
www.essvote.com	ES&S	●	●	○
everyonecounts.com	Everyone Counts	●	●	○
followmyvote.com	Follow My Vote	●	●	○
id.ee	Estonian National Identiy	●	●	○
intelivote.com	Intelivote	●	●	○
polyas.com	Polyas	●	●	○
polys.me	Polys	●	●	○
scytl.com	Scytl	●	○	○
smartmatic.com	Smartmatic	●	○	○
simplesurvey.com	Simple Survey	●	○	○
simplyvoting.com	Simply Voting	●	○	○
sk.ee	SK ID Solutions	●	○	○
tivi.io	TIVI	●	○	○
voatz.com	Voatz	◐[b]	○	○
vogo.vote	Vogo	●	○	○
votebox.co	Vote Box	●	●	○
votem.com	Votem	●	○	○
votewatcher.com	Vote Watcher	◐[a]	○	○
votingplace.net	Voting Place	◐[b]	○	○

[a] Certificate common-name mismatch with cloud host.
[b] Vulnerable to POODLE (CVE-2014-3566).

Table 4. HSTS in Australian Election Agencies

Domain	Division	TLS	HSTS	Preload
aec.gov.au	Australia	●	○	○
elections.act.gov.au	Australian Capital Territory	●	○	○
elections.nsw.gov.au	New South Wales	●	○	○
ntec.nt.gov.au	Northern Territory	●	○	○
ecq.qld.gov.au	Queensland	●	○	○
ecsa.sa.gov.au	South Australia	●	○	○
tec.tas.gov.au	Tasmania	●	○	○
vec.vic.gov.au	Victoria	●	●	○
waec.wa.gov.au	Western Australia	●	○	○

Table 5. HSTS in Canadian Election Agencies

Domain	Division	TLS	HSTS	Preload
`elections.ca`	Canada	○	○	○
`elections.ab.ca`	Alberta	○	○	○
`elections.bc.ca`	British Columbia	◐[a]	○	○
`electionsmanitoba.ca`	Manitoba	○	○	○
`www.electionsnb.ca`	New Brunswick	○	○	○
`www.elections.gov.nl.ca`	Newfoundland	○	○	○
`electionsnwt.ca`	Northwest Territories	○	○	○
`electionsnovascotia.ca`	Nova Scotia	●	○	○
`elections.nu.ca`	Nunavut	●	○	○
`elections.on.ca`	Ontario	●	○	○
`electionspei.ca`	Prince Edward Island	○	○	○
`electionsquebec.qc.ca`	Quebec	◐[b]	○	○
`elections.sk.ca`	Saskatchewan	○	○	○
`electionsyk.ca`	Yukon	○	○	○

[a] Using self-signed certificate.
[b] Using weak/obsolete ciphersuites.

According to the National Conference of State Legislators,[14] 37 US states plus the District of Columbia offered online voter registration as of 2018. We examined each of these states, and the results are given in Table 6. Only Maryland, Minnesota, Ohio and South Carolina implemented HSTS, and Florida's registration site[15] was the only one found in the preload list. Interestingly, this domain failed to meet all the preload eligibility requirements (including the use of the `strict-transport-security` header). Furthermore, Florida's site was present in the Chrome preload list *only*, meaning Firefox, Safari and Edge users would not be protected from TLS stripping.

Our investigation also revealed that the voter registration websites for Alaska, Louisiana, Massachusetts and Oregon had TLS configurations with serious vulnerabilities. We notified the affected states of our respective findings. Massachusetts promptly responded by disabling SSL 2 and 3, and we had a conference call with the state director of elections about our findings and their efforts to address them.

5 Barriers to Adoption

With only 50,000 entries on the HSTS preload list out of hundreds of millions of websites worldwide, adoption among voting sites was expected to be low. The question is: what barriers are faced by election management bodies and vendors?

[14] https://ncsl.org
[15] https://registertovoteflorida.gov

Table 6. HSTS in US Voter Registration Websites

Domain	State	TLS	HSTS	Preload
alabamavotes.gov	Alabama	●	○	○
voterregistration.alaska.gov	Alaska	◐[a,b,c]	○	○
servicearizona.com	Arizona	●	○	○
registertovote.ca.gov	California	●	○	○
sos.state.co.us	Colorado	●	○	○
voterregistration.ct.gov	Connecticut	●	○	○
ivote.de.gov	Delaware	●	○	○
vote4dc.com	D.C.	●	○	○
registertovoteflorida.gov	Florida	●	○	◐[f]
registertovote.sos.ga.gov	Georgia	●	○	○
olvr.hawaii.gov	Hawaii	●	○	○
apps.idahovotes.gov	Idaho	●	○	○
ova.elections.il.gov	Illinois	●	○	○
indianavoters.in.gov	Indiana	●	○	○
sos.iowa.gov	Iowa	●	○	○
www.kdor.ks.gov	Kansas	●	○	○
vrsws.sos.ky.gov	Kentucky	●	○	○
sos.la.gov	Louisiana	◐[c]	○	○
voterservices.elections.state.md.us	Maryland	●	●	○
www.sec.state.ma.us	Massachusetts	◐[c,d]	○	○
mnvotes.sos.state.mn.us	Minnesota	●	●	○
sos.mo.gov	Missouri	●	○	○
nebraska.gov	Nebraska	●	○	○
nvsos.gov	Nevada	●	○	○
portal.sos.state.nm.us	New Mexico	●	○	○
dmv.ny.gov	New York	●	○	○
olvr.sos.state.oh.us	Ohio	●	●	○
secure.sos.state.or.us	Oregon	◐[e]	○	○
www.pavoterservices.state.pa.us	Pennsylvania	●	○	○
vote.sos.ri.gov	Rhode Island	●	○	○
info.scvotes.sc.gov	South Carolina	●	●	○
ovr.govote.tn.gov	Tennessee	●	○	○
secure.utah.gov	Utah	●	○	○
olvr.sec.state.vt.us	Vermont	●	○	○
vote.virginia.gov	Virginia	●	○	○
wei.sos.wa.gov	Washington	●	○	○
ovr.sos.wv.gov	West Virginia	●	○	○
myvote.wi.gov	Wisconsin	●	○	○

[a] Vulnerable to CVE-2014-0224.

[b] Vulnerable to Logjam (CVE-2015-4000).

[c] Vulnerable to POODLE (CVE-2014-3566).

[d] Vulnerable to DROWN (CVE-2016-0800).

[e] Vulnerable to ROBOT (CVE-2017-13099 and others).

[f] In Chrome preload list *only*.

Education and Awareness. The first barrier predominately appears to be a lack of awareness of TLS stripping, its implications, and the action required to protect against it. Increasing awareness begins by identifying at-risk websites and engaging with them. Eventually we hope to encourage the Qualsys SSL Test to include HSTS preloading in their grade scoring.

Technical Barriers. Membership in the preload list nominally requires: a valid TLS configuration; HTTPS redirects on the same domain; HSTS header with a sufficient `max-age`; the `include-subdomains` field; and, the `preload` field. Meeting these requirements can be non-trivial within the given web environment. Many web application frameworks (e.g., ASP .NET, Django, etc.) support HSTS via 3rd party plugins. The Helios maintainers observed that while Django offered an HSTS module, it did not support the preload field. In other cases (e.g., Apache, Nginx), preloading is a non-standard option which can be included only through a custom user-defined configuration. Microsoft's IIS server explicitly supports a preload attribute as of version 10, however all the US voter registration sites we observed running IIS used versions 8.5 or below.

Institutional Barriers. It is the root domain (e.g., `example.com`) that must be added, even if it is only a subdomain requiring preloading. This can be problematic in large institutions with numerous subdomains where applying a blanket HSTS policy would cause functional issues to, e.g., development severs. For example, Culnane et al. [5] observed that the online voting website of the 2017 Western Australian state (`ivote-cvs.elections.wa.gov.au`) was being hosted on the New South Wales iVote server, but would require action from the Western Australian IT staff (`wa.gov.au`) to be included in the preload list.

Scalability. Finally, there are scalability considerations to the preload list itself. Many election URLs are active only during polling period, and some domains have a clear one-time use (e.g., `election2020.org`). Adding all election websites to the preload list would, over time, risk filling it with stale domains.

We contacted the maintainers of the Chrome preload list and they were clear that, for the time being at least, all election sites are recommended for HSTS preloading, and that they be added at least 3 months in advance of the polling period to ensure time to be pushed out to voters via browser updates.

Conclusion

If you plan to deliver election and voting services online, the best practice to prevent TLS stripping attacks is to add your domain to the HSTS preload list. All voters using an updated browser will then be directed to your site securely over HTTPS, even if someone in your organization directs them to type or click an insecure HTTP link. This paper presented a study of over one hundred websites related to online elections and found almost none are presently doing this. The reasons are varied, but predominantly seem to be a matter of a lack of awareness of this issue, and we hope this paper will aid in this regard.

Acknowledgements

Thanks to Ben Adida, Kirsten Dorey, Lucas Garron and the anonymous reviewers.

References

[1] D. Adrian, K. Bhargavan, Z. Durumeric, P. Gaudry, M. Green, J. A. Halderman, N. Heninger, D. Springall, E. Thomé, L. Valenta, et al. Imperfect forward secrecy: How diffie-hellman fails in practice. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 5–17. ACM, 2015.

[2] M. Alsharnouby, F. Alaca, and S. Chiasson. Why phishing still works: User strategies for combating phishing attacks. *International Journal of Human-Computer Studies*, 82:69–82, 2015.

[3] C. Amrutkar, P. Traynor, and P. C. Van Oorschot. An empirical evaluation of security indicators in mobile web browsers. *IEEE Transactions on Mobile Computing*, 14(5):889–903, 2015.

[4] J. Clark and P. C. van Oorschot. Sok: Ssl and https: Revisiting past challenges and evaluating certificate trust model enhancements. In *Security and Privacy (SP), 2013 IEEE Symposium on*, pages 511–525. IEEE, 2013.

[5] C. Culnane, M. Eldridge, A. Essex, and V. Teague. Trust implications of ddos protection in online elections. In *International Joint Conference on Electronic Voting*, pages 127–145. Springer, 2017.

[6] R. Dhamija, J. D. Tygar, and M. Hearst. Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 581–590. ACM, 2006.

[7] K. Dorey, N. Chang-Fong, and A. Essex. Indiscreet logs: Diffie-hellman backdoors in tls. In *Proceedings of the 24th Annual Network and Distributed System Security Symposium (NDSS 2017). The Internet Society*, 2017.

[8] Z. Durumeric, J. Kasten, D. Adrian, J. A. Halderman, M. Bailey, F. Li, N. Weaver, J. Amann, J. Beekman, M. Payer, et al. The matter of heartbleed. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 475–488. ACM, 2014.

[9] S. Dzieduszycka-Suinat, J. Murray, J. Kiniry, D. Zimmerman, D. Wagner, P. Robinson, A. Foltzer, and S. Morina. The Future of Voting: End-to-End Verifiable Internet Voting - Specification and Feasibility Study. US Vote Foundation, 2015.

[10] A. Essex. Detecting the detectable: Unintended consequences of cryptographic election verification. *IEEE Security & Privacy*, 15(3):30–38, 2017.

[11] B. Fogel, S. Farmer, H. Alkofahi, A. Skjellum, and M. Hafiz. Poodles, more poodles, freak attacks too: how server administrators responded to three serious web vulnerabilities. In *International Symposium on Engineering Secure Software and Systems*, pages 122–137. Springer, 2016.

[12] J. A. Halderman and V. Teague. The new south wales ivote system: Security failures and verification flaws in a live online election. In *International Conference on E-Voting and Identity*, pages 35–53. Springer, 2015.

[13] J. Hodges, C. Jackson, and A. Barth. HTTP Strict Transport Security (HSTS), RFC 6797, 2012.

[14] M. Kranch and J. Bonneau. Upgrading https in mid-air: An empirical study of strict transport security and key pinning. In *NDSS*, 2015.

[15] M. Marlinspike. More tricks for defeating ssl in practice. *Black Hat USA*, 2009.

[16] E. Moher, J. Clark, and A. Essex. Diffusion of voter responsibility: Potential failings in e2e voter receipt checking. In *USENIX Journal of Election Systems and Technology*, 2015.

[17] M. Nemec, M. Sys, P. Svenda, D. Klinec, and V. Matyas. The return of coppersmith's attack: Practical factorization of widely used rsa moduli. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1631–1648. ACM, 2017.

[18] E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2, RFC 5246, 2008.

[19] E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3 (Draft 28), 2018.

[20] Y. Sheffer, R. Holz, and P. Saint-Andre. Summarizing known attacks on transport layer security (tls) and datagram tls (dtls). *RFC 7457*, 2015.

[21] D. Springall, T. Finkenauer, Z. Durumeric, J. Kitcat, H. Hursti, M. MacAlpine, and J. A. Halderman. Security analysis of the estonian internet voting system. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 703–715. ACM, 2014.

[22] L. Valenta, D. Adrian, A. Sanso, S. Cohney, J. Fried, M. Hastings, J. A. Halderman, and N. Heninger. Measuring small subgroup attacks against diffie-hellman (eprint). In *Proceedings of the 24th Annual Network and Distributed System Security Symposium (NDSS 2017). The Internet Society*, 2017.

[23] L. Valenta, S. Cohney, A. Liao, J. Fried, S. Bodduluri, and N. Heninger. Factoring as a service. In *International Conference on Financial Cryptography and Data Security*, pages 321–338. Springer, 2016.

[24] Western Australian Electoral Commission. 2017 State General Election Election Report, 2017.

[25] F. Zagórski, R. T. Carback, D. Chaum, J. Clark, A. Essex, and P. L. Vora. Remotegrity: Design and use of an end-to-end verifiable remote voting system. In *International Conference on Applied Cryptography and Network Security*, pages 441–457. Springer, 2013.

PhD Colloqium

The Risk Limit of Bayesian Audits

Kellie Ottoboni[0000−0002−9107−3402]

University of California, Berkeley
kellieotto@berkeley.edu

1 Introduction

A *risk-limiting post-election audit* is a procedure for confirming the outcome of an election which has a known, pre-specified minimum chance of correcting an incorrect electoral outcome. There are two types of audits which differ in how they handle ballots: ballot comparison, which compares each paper ballot to its electronic record, and ballot-polling, which looks at paper ballots alone when ballots cannot be linked to the corresponding electronic record. The voting technology used in the election determines which type of audit is used: ballot comparison audits are the more efficient option, but can only be used with voting machines that produce a cast-vote record. Simple statistical methods have been developed for risk-limiting ballot comparison and ballot-polling audits of plurality contests and other relatively simple social choice functions ([4,5,2]).

Risk-limiting audit procedures have not been derived for more complex elections, because the statistical likelihood cannot be written in a simple way. Examples include elections where precincts use heterogeneous voting machines and elections with complex social choice functions. [3] proposed *Bayesian audits* as a solution. Bayesian audits use simulation, rather than analytical expressions, to estimate the posterior probability that the reported outcome is incorrect, given the observed data. This auditing method is agnostic to the social choice function and can work for both ballot-polling and ballot comparison audits.

In general, Bayesian audits do not control the risk of failing to correct an incorrect electoral outcome. The goal of this paper is to determine if and when Bayesian audits are risk-limiting. Understanding the relationship between Bayesian audits and risk-limiting audits may allow us to derive new risk-limiting audits for elections with complex social choice functions.

We will address two questions in this paper:

1. When is the Bayesian audit method of [3] risk-limiting?
2. Can we exploit the Bayes-minimax duality to derive new risk-limiting audits based on existing Bayesian ones?

2 Methods

The first question can be addressed through simulated audits. In simulations, the hypothetical election results are known, so we can check how frequently

Bayesian audits incorrectly confirm the outcome. If it is more often than the specified risk level, then the procedure is not risk-limiting.

Preliminary simulations have demonstrated that Bayesian audits are *not* risk-limiting: they incorrectly confirm the outcome more frequently than the desired risk limit.

The second question must be addressed with theory. *Decision theory* is a statistical framework for deriving optimal decisions under uncertainty. In election audits, uncertainty arises from the unknown true results and the sampling of ballots. Bayesian audits and risk-limiting audits model these uncertainties differently; this framework allows us to determine when the optimal auditing procedures coincide under these two models.

The *Bayes-minimax duality* states that the Bayes estimator with the least favorable prior (the one that maximizes the prior-average loss) is equivalent to the minimax frequentist estimator [1]. Therefore, if we can find the Bayes estimator and least favorable prior for an audit, then this automatically gives us a frequentist estimator with certain optimality properties. The next problem, then, will be to determine whether it is possible to characterize the least favorable prior analytically or whether it can be approximated computationally.

3 Significance

Bayesian audits are an exciting possibility for elections officials, as they are a computational procedure that can be used to audit any type of election. However, more investigation into their statistical properties is necessary before they can be put into practice. Determining if and when Bayesian audits are risk-limiting is a necessary first step. Moreover, characterizing when Bayesian audits are risk-limiting will solve an open problem in this area: how to develop risk-limiting audit procedures for more complex election scenarios and social choice functions.

References

1. Berger, J.O.: Statistical decision theory and Bayesian analysis. Springer Science & Business Media (2013)
2. Lindeman, M., Stark, P., Yates, V.: BRAVO: Ballot-polling risk-limiting audits to verify outcomes. In: Proceedings of the 2011 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections (EVT/WOTE '11). USENIX (2012)
3. Rivest, R.L., Shen, E.: A Bayesian Method for Auditing Elections. In: EVT/WOTE (2012)
4. Stark, P.: Risk-limiting post-election audits: *P*-values from common probability inequalities. IEEE Transactions on Information Forensics and Security **4**, 1005–1014 (2009)
5. Stark, P.: Super-simple simultaneous single-ballot risk-limiting audits. In: Proceedings of the 2010 Electronic Voting Technology Workshop / Workshop on Trustworthy Elections (EVT/WOTE '10). USENIX (2010), `http://www.usenix.org/events/evtwote10/tech/full_papers/Stark.pdf`

How to Solve the Transparency-Anonymity Dilemma?

Thomas Weiler
University of Applied Sciences for Public Administration and Management NRW at Cologne, Germany
thomas.weiler@fhoev.nrw.de

In March 2009 the German Constitutional Court (Bundesverfassungsgericht - BverfG) ruled that elections in Germany in addition to complying with the written rules of general, direct, free, equal and secret (Art. 38 para 1 sentence 1 of the German Basic Law - Grundgesetz - GG) also have to be transparent [1]. “Transparent” in this context means that the average voter has the right to understand the voting process and how it leads to the eventual result of the election. An average voter in this context is one without any specific knowledge in the matter [2] and no further knowledge in the field of computer science [3]. The level of understanding of the voting process does have to be deep enough for the average voter to follow the essential steps of the voting process from casting the ballot through final result in order to allow him or her to determine whether the process transpired without undue interference or manipulation [4].

This decision also decreed the public accessibility of the voting process. The “Public” in this context encompasses all eligible voters [5]. As elections and referenda have to be public and transparent throughout, there needs to be a possibility of the voter to also supervise the counting of the ballots [6], even at a later stage. The voting machines used at the time did not print out any individual ballot papers but only stored votes cast electronically, with the final result then shown on a screen. This method was deemed insufficient by the BVerfG.

As transparency and publicity were not upheld with the voting machines used in the general election of 2005 the BVerfG declared their use and the underlying law (Bundeswahlgeräteverordnung) as unconstitutional. Even though this decision only dealt with voting machines used in a supervised manner at ballot stations in the years prior, it also covers elections or referenda where remote unsupervised electronic voting (Online-voting/Internet-voting) would be used.

While there are electronic methods to ensure the vote is public and transparent, these cannot at the same time ensure the secrecy of the individual vote, i.e. the vote can either be transparent and public or anonymous (“transparency-anonymity dilemma”). So far, no electronic method exists to solve this conundrum, meaning that paper ballots are the only way to conduct voting (“paper-trail”).

Thus, electronic voting methods that do not allow for paper ballots cannot be legally/ constitutionally conducted in Germany. How can this dilemma be solved? Can an electronic method be found that allows for simultaneous transparency and secrecy of the vote? Can other principles be envisaged, e.g. substituting certification and verification methods [7] for the transparency of the vote (at the time of the BVerfG the methods used were deemed insufficient)? Can thrust-building measures be a way to make transparency unnecessary? Is there a way allowing for Online-voting including a printed ballot paper where the voter casts the vote via the internet and this very vote is submitted anonymously and then the ballot is printed out at a ballot station? Could a Voter Verifiable Paper Audit Trail (VVPAT) or Verifiable Paper Record (VPR) be a substitute? Can ways be found to explain cryptology methods in such a way that the "average" voter understands them so they can be used while upholding the transparency required to avoid manipulations? Is there a method to verify the vote is cast as intended and recorded as intended that does not provide the voter with a receipt that could be used to sell the vote? Solving this dilemma requires an interdisciplinary approach.

References

1. Decision 2 BvC 03/07 and 04/07 of 03.03.2009, BverfGE 123, 39pp., available online via juris (in German), https://www.juris.de/jportal/index.jsp, last accessed 07/20/2018.
2. Decision 2 BvC 03/07 and 04/07 of 03.03.2009, BverfGE 123, 39pp., available online via juris. https://www.juris.de/jportal/index.jsp, in juris No. 118, last accessed 07/20/2018.
3. Ibid., in juris No. 118.
4. Röckinghausen, Marc (2014) In: Sensburg, Patrick Staats- und Europarecht, Verlag für Verwaltungswissenschaft, Frankfurt/Main, p 85 (in German).
5. Bremke, Nils (2004) Der Grundsatz der Öffentlichkeit der Wahl und Internetwahlen,. In: MultiMedia und Recht, pp IX – XIII (in German).
6. Decision 2 BvC 03/07 and 04/07 of 03.03.2009, BverfGE 123, 39pp., available online via juris. https://www.juris.de/jportal/index.jsp,, in juris No. 106.
7. Volkamer, M, Schryen, G, Langer, L, Schmidt, A, Buchmann, J (2009) Elektronische Wahlen: Verifizierung vs. Zertifizierung. In: Forschungsbericht des CASED der Universität Darmstadt. https://epub.uni-regensburg.de/21295/, .last accessed 07/20/2018 (in German).

Formal Verification of Selene using the Tamarin prover

Marie-Laure Zollinger

University of Luxembourg
marie-laure.zollinger@uni.lu

Introduction

Formal verification of voting protocols in the symbolic model is a subject that is already explored in the literature. Many tools exist to perform automated verification, such as ProVerif, DEEPSEC, or TAMARIN. These tools aims to create proofs for security properties such as *Privacy*. However, each tool has different constraints to describe and execute protocols (e.g. unbounded number of sessions, false attacks in trace equivalence, or a limited number of cryptographic primitives). In TAMARIN, security proofs are modeled as trace properties. For *Privacy*, we use trace equivalences, where the adversary can't distinguish between two swapped votes. A formal model of Selene has already been performed with TAMARIN [1], which gave an automated proof for Vote-Privacy and Receipt-Freeness of the protocol. However, TAMARIN does not model some of the cryptographic primitives used in Selene, that did not allow the authors to use a standard version of the scheme. Indeed, they implemented a simplified version on the protocol, with a single Tally Teller (no distributed trust) and additional trust assumptions. In this study, the idea is to improve the model developed for Selene by using the new XOR operation developed in [3], in association with the existing rules for Diffie-Hellmann equational theory [4].

Formal description of Selene

The Selene mechanism gives to the voter a private tracking number, used to directly check her vote by consulting the final tally board. This tracker is first given to the voters after all votes and trackers have been published.

Selene uses ElGamal encryption, that is homomorphic and can act as a commitment scheme. Every voter V_i has a public key $pk_i = g^{x_i}$ where x_i is the secret trapdoor key. The election key pk_T is used to encrypt votes and trackers. A subset of t Tellers T_j perform distributed threshold decryption. A Mixnet perform distributed re-encryption of votes and trackers. Selene's workflow is as follows:

Setup Tracking numbers are generated, encrypted with the election key then mixed before being associated to a voter's key. Before the elections start, the tracker is transcrypted to an ElGamal encryption under the voter's key, that we can write as a pair (α, β). The β-term is displayed on the bulletin board, as a trapdoor commitment, the α-term is kept secret (and shared between Tellers $\alpha = \alpha_1 \cdot ... \cdot \alpha_t$).

Voting Each voter sends a signed and encrypted vote to the bulletin board, which checks the signature and displays the encrypted vote.

Tallying At the end of the elections, votes and associated trackers are extracted, mixed in parallel and threshold decrypted to be displayed on the bulletin board.

Verifying The α_i are sent to the voter, who can build her encrypted tracker by putting it together with the public commitment β-term. She decrypts it to retrieve the tracker, and verify her vote.

The interesting idea behind the scheme is the possibility for every voter to fake the tracker if demanded by a coercer. The β-term commitment can open a different value that the one initially committed using the trapdoor key.

To develop a formal model of Selene in Tamarin, we need to describe a threshold and distributed encryption scheme, as well as a commitment scheme.

XOR operation and Diffie-Hellmann exponentiation The equational theory for Diffie-Hellmann described in [4] is defined as follow:

$$(x\,\hat{}\,y)\,\hat{}\,z = x\,\hat{}(y * z), \qquad x * (y * z) = (x * y) * z, \qquad x * y = y * x$$

$$x * 1 = x, \qquad x * \mathrm{x}^{\text{-}1} = 1, (\mathrm{x}^{\text{-}1})^{-1} = x, \qquad x\,\hat{}\,1 = x$$

This equational theory does not support protocols that perform multiplication: addition of exponents must be modeled. For example, with this equational theory, we can't write: $g^a \times g^b = g^{a+b}$.

On the other hand, the XOR operation [3], defined as follow:

$$x \oplus x = 0 \qquad x \oplus (y \oplus z) = (x \oplus y) \oplus z \qquad x \oplus 0 = x$$

$$x \oplus y = y \oplus x \qquad x \oplus x \oplus y = y$$

To implement Selene in the symbolic model, we want to use the XOR operation to model a receipt that could be made from several authorities. As described before, the α-term is shared between Tellers. With 2 Tellers, we have: $\alpha = \alpha_1 \cdot \alpha_2$, that could be modeled as $\alpha = \alpha_1 \oplus \alpha_2$.

On the other hand, Diffie-Hellmann exponentiation can be used to model terms such as commitments.

Commitment schemes In addition of the previous theories, we also need a theory that includes a trapdoor commitment scheme for the trackers notification. This has been developed in [2]. An extension of Tamarin has been proposed to support additional convergent theories, i.e. that are confluent and terminating. This allowed the authors to define a new convergent theory for the trapdoor commitment scheme, that is defined by the functions *open*, *commit*, $fake$ and by the following equations:

$$\begin{aligned} open(commit(m, r, td), r) &= m \\ commit(m_2, fake(m_1, r, td, m_2), td) &= commit(m_1, r, td) \\ open(commit(m_1), r, rd), fake(m_1, r, td, m_2)) &= m_2 \\ fake(m_1, fake(m, r, td, m_1), td, m_2) &= fake(m, r, td, m_2) \end{aligned}$$

Work in progress

The next steps are to start an implementation with Tamarin that includes the forementioned equations. Then we can see if the Receipt-Freeness property that has been proved in [1] still holds.

References

1. Bruni, A., Drewsen, E., Schürmann, C.: Towards a mechanized proof of selene receipt-freeness and vote-privacy (2017)
2. Dreier, J., Duménil, C., Kremer, S., Sasse, R.: Beyond subterm-convergent equational theories in automated verification of stateful protocols (2017)
3. Dreier, J., Hirschi, L., Radomirovic, S., Sasse, R.: Automated Unbounded Verification of Stateful Cryptographic Protocols with Exclusive OR (2018)
4. Schmidt, B., Meier, S., Cremers, C.J.F., Basin, D.A.: Automated analysis of diffie-hellman protocols and advanced security properties (2012)

Why is paper-based voting the most secure option?

Tamara Finogina

UPC & Scytl Secure Electronic Voting, Spain
tamara.finogina@scytl.com

It has been noted that the main purpose of the election is not simply select a winner, but to convince the losing party that they lost [2]. The paper-based election leaves an auditable trail that allows performing an arbitrary number of re-counts in case of doubts. This paper trail is assumed to be set in stone and should guarantee that honest re-count would reveal the true voters intent.

However it is often forgotten that traditional paper-based voting scheme is not secure on its own. Over the course of years, the election procedure has been changed multiple times to ensure the highest standards of security possible. For example, until 1990s elections were not commonly supervised, while nowadays observers participation is considered to be desirable or even mandatory. Nevertheless, the underline construction: “every voter writes his intent on an secure paper ballot and puts it in a box, then all ballots are counted manually” provides voter privacy, but no integrity. Even if an adversary substitutes every single ballot in that box, the public will never know.

We can think of two main reasons why traditional voting system worked so far: first, it is **distributed** and second – the existence of **additional security measures** such as observers, watermarks on ballots, physical presence, etc.

The first mechanism – unavoidable physical decentralization of polling stations – intends to minimize the overall damage in case an adversary was able to corrupt some stations. Even though it is still possible to remove, modify or insert ballots and therefore change the election results in some polling places, for affecting the entire election outcome, the adversary has to control many stations, significantly increasing the probability of being caught with every corruption.

Additional security measures, on the other hand, aim to protect ballots integrity by making it difficult for an adversary to gain unsupervised or unauthorized access or at very least increasing the cost of such attacks. Some mechanisms intend to prevent ballot replication (stamps, watermarks, a specific type of ballot paper, etc), others focus on ensuring voters eligibility (physical presence, identity verification, voters registration, etc) or providing transparency (observers, public counts, CCTV cameras, etc). The general rule that brings some degree of transparency to the traditional paper-based voting is the following: all operations should be done in presence of observers and no one should have access to the ballots without supervision. Nevertheless, all those measures serve the same goal - ensure that people responsible for registering voters, storing ballots, counting and announcing the tally behave.

Today electronic voting is commonly accused of being insecure by design. Security experts warn the public against the danger of using malicious hardware, software, and centralized systems in elections [1]. However, it is often neglected that e-voting cannot be secured by simply applying the same policy adopted for traditional paper elections [3].

E-voting has its strengths and weaknesses that are far too different from those of traditional elections. Unlike the traditional approach, that mostly focuses on privacy, e-voting was developed to provide verifiability. Indeed, in contrary to the traditional paper-based voting where physical presence and coordination of auditors are necessary, for the e-voting system a single auditor can perform verification of 100% of ballots multiple times. Moreover, e-voting enables repeated audit of operations that previously were not verified (e.g. eligibility of votes) or could be checked only once (e.g. unsealing envelopes). On the other hand, e-voting is inherently centralized so the damage of a successful attack can be far more significant. Also, the usability that remote e-voting brings comes at a price of extending possible attack scenarios as now an attacker is not limited to a physical location. At the same time, e-voting scales and offers verifiability and integrity that are based on mathematical assumption rather than on the presence of observers. Furthermore, although privacy is considered to be one of the main features of paper-voting, it is not technically accurate due to a possibility of tracking ballots based on unique identifiers or even slight variations in color and 3D surface texture of paper [3]. By contrast, digital ballots can be shuffled and re-encrypted in a verifiable manner to avoid tracking.

Overall, the traditional paper-based voting is far more resistant to the tally manipulation attacks than e-voting solutions. However, it is a lot easier to detect those attacks for e-voting, than for the traditional approach.

Nowadays paper-based voting is our gold standard for truly trustworthy elections. The combination of years of exploitation and the intuitive simplicity of the election process makes the traditional voting the most trusted option in public eyes. However, it was not always as secure as we believe it to be. People made it secure by developing strict election procedures and policies. Without those measures, the paper-based election is basically a leap of trust.

Some argue that e-voting is inevitable, others try to ban it as a potential threat to democracy and our free will. While both sides have strong arguments, they often compare e-voting in abstract and the traditional paper-voting with all its historically established security measures. Thus, one question still remains open: *If we were to develop some additional procedures how then e-voting be different from paper-based elections?*

References

1. Halderman, J.A.: Practical attacks on real-world e-voting. In: Hao, I.F., Ryan, P.Y.A. (eds.) Real-World Electronic Voting: Design, Analysis and Deployment, chap. 7, p. 145171. CRC Press, Oxford (2004)
2. Schneier, B.: Blog post: Securing elections . https://www.schneier.com/blog/archives/2018/04/securing_electi_1.html
3. Willemson, J.: Bits or paper: Which should get to carry your vote? J. Inf. Sec. Appl. **38**, 124–131 (2018)

Blockchain Technology for End-To-End Verifiable Elections on Internet Voting System

The Estonian Case-study

Rodrigo Cardoso Silva[1 2] [0000-0002-2702-7137]

[1] Pontifical Catholic University of Sao Paulo, Caio Prado street, 102 São Paulo, Brazil
[2] Tallinn University of Technology, Akadeemia tee 3, 12618 Tallinn, Estonia
rodrigo.pucsp@outlook.com

Remote state-citizen communication has been implemented in many communities, but Estonia has been one of the most eager countries to actively pursue electronic services and procedures. Estonia has featured a remote online voting method since 2005, and has been the only country in Europe (not to say the world) to have it with-out limitations in all types of elections. Theoretical literature in the computer science is often related to voting from an uncontrolled environment and connected technical risks (e.g. security of the voting device and voting channel). Most of the papers and new scientific thought are being channeled to the vision of finding the safest, tamper-proof, mathematically sound system currently possible. This field of study looks for the ideal solution to answer all possible theoretical risks and practical acceptance and many articles are devoted to a topic that has been seen as the number one confidence builder in remote Internet Voting systems – verification (individual, universal or in multiple stages)1 [1].

Estonia has implemented the recorded as cast level in 2013; however, discussions about possible additional steps in this field are ongoing. The verification scene is very rich and filled with different ideas to offer credible ways towards higher verifiability[2].

In 2008, blockchain gave rise to a new model of information security on the Internet and has become an emergent topic[3].

Last year, the Estonian government had a major problem with the security of ID cards [3]. This fact increased the use of Smart-ID and Mobile-ID by Estonians [4].

In this research, we aim to develop verifiable end-to-end voting via blockchain on internet voting system. The research question is: a) How to implement the blockchain

[1] Auditing elections with the use of encryption isn't a new subject. Since 2004, there are several systems that allow voters to use verifiable end-to-end tools (E2E), for instance, visual cryptography, Punchscan system, Prêt à Voter system, Scratch and Vote system, Three Ballot voting protocol, Scantegrity (I and II) systems and STAR-Vote system.

[2] Historically, in the early 2000s, the domain of trust building in (remote) electronic voting solutions was dominated by the concept of certification. Over the years, and with the growing possibilities of different solutions, verifiability has grown to be the main factor in guaranteeing the theoretical trustworthiness of an electronic voting solution [1].

[3] We can consider that blockchain 1.0 was for the decentralization of money and other forms of payments; while blockchain 2.0 is being used for the decentralization of markets in a more general way. Today, technology includes the transfer of many other types of assets that go beyond currency, like voter's registers [2].

in internet voting (i-voting) on the X-Road system? b) How can we prevent the cost of the election from becoming more expensive due to the use of the blockchain? c) How to prevent X-Road system performance from losing performance?

According to the Estonian government, blockchain has been testing the technology already since 2008 [5]. Furthermore, the X-Road system is perfect for this research because is distributed data system.

1 How could this be done?

Firstly, it is important to understanding that, in theory, the blockchain will be implement on the security layer of the X-Road system. In this way, the system maintains its decentralized data system structures. Secondly, the challenge of this research isn't the encryption, but the engineering, because it's necessary to build a scalable and reliable service for the government without any kind of interruption. And third, the main principle is the validation of each voter's vote4.

Regarding the scientific contribution of this research, it will add significantly increase the security of Internet voting in Estonia (possibly in other countries) and empowering voter activism[5].

References

1. Vinkel, P., Krimmer, R.: The How and Why to Internet Voting an Attempt to Explain E-Stonia. In:1st International Joint Conference, E-Vote-ID 2016, pp. 178-191. Bregenz, Austria, October 18-21 (2016).
2. Swan, M. Blockchain: Blueprint for anew economy. O'Reilly Media, Inc., pp. 12, 2015.
3. REPUBLIC OF ESTONIA. Information System Authority. ROCA Vulnerability and eID: Lessons Learned (2018).
4. GEENIUS. The introduction of Smart-ID and Mobile-ID has increased six times after ID-card problems. https://geenius.ee/uudis/id-kaardi-sertifikaatide-pani-inimesed-smart-id-ja-mobiil-id-jareletormi-jooksma/ (2017)
5. E-ESTONIA 2018. https://e-estonia.com
6. RE:PUBLICA 2017. Digital Democracy: E-Voting for everyone? (2017) https://www.youtube.com/watch?v=EmpGXcFl9PY

[4] In theory, the I-voting would be as follows: 1) The vote of each voter shall be dismembered by the I-voting system; 2) Each piece would be encrypted by the i-voting system to maintain the anonymity, secret of vote and integrity of the vote with security; 3) The parts of each vote would be distributed through the I -voting system between the different computers of the voters that would make the storage of the files in the own computer. Unlike the bitcoin process, here there is no data mining or any other kind of "prize". Soon, the computers of the voters would automatically agree to the distribution of the different encrypted files; 4) After the voter's personalized access to the i-voting system, the system would gather the votes through the system to decrypt it; 5) The voter has the power to visualize the vote and validate it.

[5] In 2017, Priit Vinkel, Chancellery of Riigikogu, Head of State Electoral Office of Estonia, participated the Re:publica 2017 Conference and confirmed that it is possible to use the blockchain on internet voting system in Estonia [6].

The impact of I-voting on conventional voting channels

Iuliia Krivonosova [1[0000-0001-7246-1373]]

[1] Tallinn University of Technology, Tallinn, Estonia
iuliia.krivonosova@ttu.ee

1 Introduction

Remote I-voting is most frequently introduced as an additional voting channel to existing voting infrastructure (like in cases of Australia, Canada, Estonia, France, the Netherlands, Switzerland, the UK and the USA [1]). Therefore, I-voting as any other alternative voting channel should be integrated into the existing electoral infrastructure in order not to collude with other voting channels and not to challenge the overall integrity of elections [2]. Hence, introduction of I-voting as an alternative voting channel requires process reengineering from electoral administration. In this paper, we trace what processes of election delivery are reengineered with the introduction of I-voting, and whether and how introduction of I-voting impacts paper-based voting channels.

Thus, the research questions are:

1. What processes of election delivery are reengineered with introduction of I-voting?
2. What impact I-voting has on delivery of conventional voting channels?

The relevance of this research lies in the importance of organizational aspect of I-voting implementation which frequently remains overlooked in the research on I-voting. Electoral administrations all around the world are facing new challenges and new responsibilities which they need to address at the very high level to guarantee integrity of elections, despite the limited resources and skills they possess.

2 Theoretical Framework and Research Methodology

To answer these research questions, we apply the theoretical framework of business process reengineering (BPR) which has been frequently applied to analysis of public administration transformations, in general [3], but there are not that many examples of BPR application to the field of elections, besides Xenakis and Macintosh research [4,5] applying BPR framework to electoral processes redesign in the UK.

With the help of the Business Process Model and Notation software, we map processes involved into delivery of multi-channel elections to see how I-voting impacts the existing electoral infrastructure: what additional activities delivery of I-voting requires from the actors involved into delivery of paper-based voting, whether I-voting is integrated into existing electoral infrastructure and does not collude with other available voting channels, and whether points of potential optimization and process reengineering are visible.

We gather information on processes by legal analysis (considering electoral law regulating what activities should take place during delivery of elections and in what order). Then, we complement the data with on-site observation and interviews with stakeholders to see how legally defined activities are implemented in practice. For I-voting, we also derive information from log files of I-voting system as it automatically registers with timestamps all activities happening.

We apply case-study methodology [6] and consider Estonia as a critical case [7], because the country has been implementing I-voting for more than a decade in the electoral environment with a high number of alternative voting channels, all of which should be harmonized and not collude with each other.

3 Findings

Process mapping of electoral activities taking place in Estonian local elections in 2017 reveals that many processes have not been adjusted yet to the new reality of e-enabled elections despite 13 years since I-voting implementation. For instance, merging of election day voter lists with I-voter lists and advance voter lists is still done manually with many resource-consuming activities required, like transportation of those lists from Tallinn to every polling station. The steadily growing number of advance and I-voters only increases the workload. Another finding is that no process redesign or relocation of staff and resources have been done to adjust to the increasing demand for I-voting and consequently decreasing demand for paper-based voting. Recommendations on how these processes might be reengineered to guarantee less workload for Electoral Management Bodies delivering paper-based voting are developed and presented in the paper.

References

1. Barrat, J., Goldsmith, B., & Turner, J. (2012). International experience with E-voting. International Foundation for Electoral Systems.
2. Xenakis, A., & Macintosh, A. (2004). Levels of Difficulty in Introducing e-Voting. Electronic Government, 116-121.
3. Weerakkody, V., Janssen, M., & Dwivedi, Y. K. (2011). Transformational change and business process reengineering (BPR): Lessons from the British and Dutch public sector. Government Information Quarterly, 28(3), 320-328.
4. Xenakis, A., & Macintosh, A. (2006). A Generic Re-engineering Methodology for the Organized Redesign of the Electoral Process to an E-electoral Process. Electronic Voting, 86, 119-130.
5. Xenakis, A., & Macintosh, A. (2005). Using business process re-engineering (BPR) for the effective administration of electronic voting. The electronic journal of e-government, 3(2), 91-98.
6. Yin, R. K. (2009). Case study research: Design and methods (applied social research methods). London and Singapore: Sage.
7. Flyvbjerg, Bent. Five misunderstandings about case-study research. Qualitative inquiry 12.2 (2006): 219-245.

The analysis of the electoral turnout of non-electronic and electronic voting in Ecuador

Tania Ernestina Erazo Villacres

[1] Tallinn University of Technology, Tallinn, Estonia
taeraz@ttu.ee

1 Introduction

In February in 2013, Ecuadorian government decided to improve voting procedure in the country (Pozo, 2014). The decision was part of a governmental strategic plan to modernize public administration and, at the same time, to implement the new conceptualization of the state (Senplades, 2009). Unlike the previous attempts of implementing electronic voting in Ecuador[1], **Azuay[2]** [in the southern part of Ecuador], and **Santo Domingo de los Colorados[3]** [in the center of the country] became the first two provinces to vote exclusively by electronic manner on February 23th, 2014. In addition to the two provinces, a small countryside locality of the province of Pichincha, **La Morita[4]** was also allocated to vote electronically.

Although each locality had a specific hardware and software from three different countries: Azuay: Argentinian's, Santo Domingo: Venezuelan's and La Morita: Russian's, this paper will focus on the province of Azuay, whose system was Radio-frequency identification (RFID) (Pozo, 2014). RFID kept the votes and then, they could be printed. The ballot contained a QR code as well, and it could be used to verify, on one hand, whether the vote was correct [citizen's verification] and on the other hand, political organizations could keep quick records on the results. According to OCED classification, this type of voting implemented in Ecuador could be classified as ballot scanning technology[5] (OCED, 2013).

Based on the turnout of non-electronic and electronic voting process carried out in the local elections on February 23th, 2014, the aim of this research is to assess whether there were any effects between the turnout of non-electronic voting and electronic voting in Ecuador. If there were, what they are; what effects electronic voting brought. The analysis of comparison will focus on the results of the elections in the province of Azuay (electronic voting process) as well as the province of Chimborazo (non-electronic voting process). Two comparable provinces.

[1] Ecuador implemented electronic voting in 2004, and 2012 (Villacís, 2011).

[2] 609 000 voters in Azuay.

[3] 300 000 voters in Santo Domingo de los Colorados.

[4] 98 000 voters in La Morita in the province of Pichincha.

[5] Ballot scanning technology uses a ballot paper that is either marked by a voter him or herself or with assistance of a ballot marking device in a polling station, which is them inserted into a scanning device and counted by electronically 'reading' the voter's mark on the ballot.

2 Theoretical Framework and Research Methodology

To answer these questions, it will be used one of the most well-known theoretical framework for assessing Information System [IS] is content, context and process [CCP]. Introduced in this field by Symons in 1991 (Stockade, 2005) and developed by other authors such as Huerta (1999), Stockdale (2005) among others, CCP allows researcher to explore in a broaden rage of influences, by characteristically including social, political, cultural and economic aspects.

The CCP framework has three main components:

Content – "what" is being evaluated
Context - "why" and "who" evaluate IS implementation
Process – "how" and "when" evaluation is being done.

As mentioned previously, the turnout of the local elections of February in 2014 will be taken as a case for the case study methodology; the paper will refer to Robert Yin's works. Yin (2003) pointed out that "the case study, like other research strategy is a way of investigation an empirical topic by following a set of per specified procedure". In addition to this, from the comparison of the results of the electoral process, it is expected that the electronic voting process has enfranchised different societal groups as well as reduced the absenteeism of voting.

References

1. Huerta, E. and Sanchez, P.J., (1999), *Evaluation of information technology: Strategies in Spanish firms*, European Journal of Information Systems 8, 273–283.
2. OSCE/ODIHR (2013), *Handbook for the Observation of New Voting Technologies,* published by the OSCE Office for Democratic Institutions and Human Rights (ODIHR), Warsaw, Poland.
3. Pozo, J, (2014), *Implementation Project Electronic Voting Azuay 2014 – Ecuador*, in 6th Conference of Electronic Voting, E-vote.
4. Senplades (2009), *Plan Nacional para el Buen Vivir, 2009 – 2013,* Ecuador.
5. Stockdale R, and Standing, C (2005), *An interpretive approach to evaluating information systems: A content, context, process framework*, European Journal of Operational Research.
6. Villacís, Nubia (2014), *Voto electrónico en Ecuador, un reto cumplido y la Puerta a la automatización electoral,* in *Elecciones*, ONPE, Vol 13, No 14, enero-diciembre, 2014.
7. Yin, Robert (2003), *Introduction*, in *Case Study Research: Design and Methods".* Thousand Oaks, Cal.: Sage Publications, pp. 1-17.

Demo Session

ASTRES - Auditable Secure Transparent and Reliable Elections System

Edouard Cuvelier, Olivier Pereira

Université Catholique de Louvain - ICTEAM/ELEN/Crypto Group
Place du Levant 3, 1348 Louvain-la-Neuve
edouard.cuvelier@uclouvain.be
olivier.pereira@uclouvain.be

ASTRES is a functioning prototype based on the open design of STAR-Vote [1]. ASTRES is made of three components: a voting booth, a public bulletin board and a voting urn. The three components are physically independent and while the voting booth and the urn are fixed pieces of furniture, the public bulletin board is a server hosting a website.

ASTRES targets governmental elections where the voters must go in-person to polling stations. In those stations, after id authentication, voters are directed to isolated voting booths where they produce their electronic ballot as well as their paper audit trail. The paper audit trail is composed of two parts: one that reproduces the choices of the voters and one that contains a commitment to the vote. The voter places the paper reproduction of his choices into the voting urn. This action has the effect of casting the corresponding electronic ballot. Then the voter takes the paper containing the commitment back home.

ASTRES is suited for large-scale elections with any type of tallying function (referendum, ranked voting, etc.).

ASTRES is designed to be user-friendly for person with disabilities. In particular the voting booth of ASTRES can be such that people with reading/hearing difficulties can vote with enough comfort.

ASTRES offers universal verifiability in the sense that

1. a voter can verify that his vote is cast as intended through the use of the paper audit trail and Benaloh challenge;
2. a voter can verify that his vote has been taken into account in the result of the elections thanks to the public bulletin board and the cryptographic proofs;
3. a voter/observer can have a pretty strong guarantee that the result of the election is within a margin of error with respect to the true result thanks to the statistically audited paper urns.

ASTRES guarantees perfect privacy for the voters as long as no eavesdropping occurs during the voting process and the elections decryption keys are not leaked or misused. It is worth to notice that the bulletin board contains only perfectly hiding information on the votes.

ASTRES is currently a prototype and has not been deployed in real elections yet.

1. Bell, S., Benaloh, J., Byrne, M.D., DeBeauvoir, D., Eakin, B., Fisher, G., Kortum, P., McBurnett, N., Montoya, J., Parker, M., et al.: Star-vote: A secure, transparent, auditable, and reliable voting system. USENIX Journal of Election Technology and Systems (JETS) 1(1) (2013) 18-37

The Digital-2-Paper Hybrid Model Mobile Voting System

Matthew Heuman[0000-0002-6450-2515]

#3 30 Bryan Ct. Kitchener, ON, N2A4J5
matthew@neuvote.com

The digital-2-paper (D2P) method of mobile voting addresses the con-cerns around online voting systems by connecting voters to the traditional paper ballot via mobile devices using visual identity confirmation and visual ballot con-firmation. This system implements a hybrid model approach to voting by adopt-ing the best-of-both-worlds concept by concluding that human backed identity verification is needed as well as visual verification of the paper ballot for voter confidence. The system utilizes modern communication technology enhance-ments to facilitate this connection that is restricted to mobile devices.

1 System Requirements and Properties

The D2P system utilizes the accessibility and advantages of smartphone technology adoption to augment the traditional method of paper based voting by connecting voters to a paper ballot receiver using end-to-end encrypted video communication protocols. This enhancement of typical software systems provides both tactile and sensory assur-ance to a voter that their vote is marked as intended and confirmed in a way that is familiar to voters from all democratic electoral systems. By storing the vote in an analog format such as the paper ballot, this also removes the data security concern from typical software based online voting systems. The D2P system stores no data in digital format, only moves it in transit while the vote is occurring. This data security method eliminates many typical attack vectors and also provides enhanced voter and ballot secrecy from being exposed. End-to-end encryption with man-in-the-middle mitigation also provides network obfuscation and observance. This system can be used in both controlled and uncontrolled environments as tablets can replace voting machines in polling locations and also voter accessible via mobile devices. The design facilitates all types of ballot based elections including binary ballots, multi-riding, referendums, proposition & non-governmental. The D2P system augments traditional voting as much as possible includ-ing providing provisions for voters with disabilities ease-of-access to their ballot. Indi-vidual and universal verifiability are maintained via end-to-end visual communication technology & confirmation code for administrator auditability as well as allowing full paper ballot recount if needed. The system has been tested in-house and is currently undergoing proof-of-concept testing in non-governmental elections. For more infor-mation please visit www.neuvote.com and download the Digital-2-Paper Concept White Paper.

Online voting verifiability and security on Scytl's voting system

Scytl Secure Electronic Voting
08008 Barcelona, Spain
www.scytl.com

Scytl's online voting system has been a pioneer in the introduction of verifiability in online voting schemes for political elections. Starting from 2004 in Switzerland (Neuchâtel), Scytl's voting system included voting receipts1, allowing voters to check that their vote was present in the final tally. In Norway, in 2011 and 2013, Scytl's online voting system introduced individual verifiability for the first time in a national election using return codes[2], and counted-as-recorded verifiability using universal verifiable Mix-nets[3,4]. In 2015, Scytl's voting system implemented a second verification mechanism designed for the State of New South Wales (Australia), based on a cast and decrypt approach (decryption of the vote in a trusted environment accessible by phone)[5]. Also in 2015, Scytl's individual verifiability (return codes) was adopted in Switzerland (Neuchâtel) and achieved in 2017 the Swiss certification for individual verifiable systems[6]. Scytl online voting system is currently in the process to achieve the Swiss complete verifiability certification level. Recently, Scytl's online voting system has been selected again by the State of New South Wales to implement a universal verifiable Mix-net and an improved cast-and-decrypt individual verifiability approach.

In the demo session, Scytl will explain the different verification mechanisms implemented by its online voting systems, the certification processes achieved by the system (including provable security) and which additional functionalities are supported in addition to verifiability, like homomorphic tally or blockchain integration[7].

[1] Puiggalí, J., Morales-Rocha, V.: Independent voter verifiability for remote electronic voting. In: Proceedings of International Conference on Security and Cryptography (SECRYPT '07), pp. 333–336, Barcelona (2007).

[2] Puiggalí, J., Guasch, S.: Universally verifiable efficient re-encryption mixnet. In: Electronic Voting 2010 (EVOTE 2010), 4th International Conference, LNI, vol. 167, pp. 241–254, Austria (2010).

[3] Puiggalí, J., Guasch, S.: Cast-as-intended verification in Norway. In: 5th International Conference on Electronic Voting 2012, (EVOTE 2012), LNI, vol. 205, pp. 49–63, Austria (2012).

[4] Wikström, D.: A sender verifiable mix-net and a new proof of a shuffle. In: Proceedings of the 11th International Conference on Theory and Application of Cryptology and Information Security. pp. 273–292. ASIACRYPT'05, Springer-Verlag, Berlin, Heidelberg (2005).

[5] Brightwell, I., Cucurull, J., Galindo, D., Guasch, S. An overview of the iVote 2015 voting system. Tech. rep. New South Wales Electoral Commission (2015).

[6] Swiss Post: Audit certificates and reports. Available at https://www.post.ch/en/business/a-z-of-subjects/industry-solutions/swiss-post-e-voting , last accessed 2018/09/17

[7] Cucurull J., Puiggalí J.: Distributed Immutabilization of Secure Logs. In: Security and Trust Management. (STM 2016). LNCS, vol 9871. Springer, Cham. Greece (2016).

A mobile application for Selene e-voting protocol

Marie-Laure Zollinger

marie-laure.zollinger@uni.lu

We present a mobile application which is a user interface of Selene e-voting protocol.

The unique idea of Selene's mechanism is that it lets voters in a given election verify their individual votes using a tracking number, after the election outcome has been published. The environment envisaged is remote internet voting.

This application has been developed in collaboration with a HCI team with interactive inputs from UX experts. The app has been used for user tests in a lab, in order to evaluate the impact of the verification phase and the impact of communicating security features on the user experience.

The context described to the participants of the test are the national elections in France. The goal was to give them high stakes, for a better self projection. Participants were selected via social networks on community groups from French cities close to the University of Luxembourg. The study has 38 participants; 24% of 18-24, 29% of 25-34, 24% of 35-44, 13% of 45-54 and 11% of 55+; 32% have a A-level, 24% have some college degree, 13% have a Bachelor's degree, 16% have a Master's degree, 3% have a PhD, and 13% have no diploma.

As mentioned above, the application implements an interface for Selene, that provides a tracking number to voters to let them verify their vote. This protocol provides receipt-freeness and coercion mitigation, by letting a voter choose another tracking number to give to a potential coercer. For this study, the application was focused on the verifiability aspects of the protocol. The application has been developed to help the user understand the purpose of individual and universal verifiability. Indeed, these steps are not commonly performed, and being able to see a list of votes might be troublesome if no further explanation is provided.

In addition, for E-Vote-ID demo session, a new version of the application has been developed that provides the coercion mitigation feature. It allows the voter to request a fake tracking number, in case of a coercer asked for it. As a consequence, the voter won't be able to verify her own vote but will be provided only with this fake tracker, i.e. coerced voters loose their right to individual verification.

The next step will be to test this new feature and to provide a complete implementation by combining our interface with our already developed library of cryptographic primitives used in Selene.