TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technology
Department of Software Science

ITC70LT
Chengxiang Wang 132120IVCMM

CLASSIFICATION OF BLACK-BOX SECURITY REDUCTIONS AND ORACLE SEPARATION TECHNIQUES

Master's thesis

Supervisor: Ahto Buldas, Ph.D

Tallinn 2017

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Chengxiang Wang

May 18, 2017

Abstract

For nearly five decades, black-box reductions are the common technique to prove the security of cryptographic constructions based on other cryptographic primitives. Informally, a black-box reduction of a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$ is a construction of $\mathcal{P}$ out of $\mathcal{Q}$ that does not use any internals of the primitive $\mathcal{Q}$ except the input and output behavior. However not all cryptographic primitives can reduce to other cryptographic primitives. As the opposition of black-box reductions, black-box separation results also have been widely used to argue the infeasibility of constructing certain cryptographic primitives.

Reingold et al. were the first to provide a widely adopted framework, called the RTV framework, to classify and relate different notions of black-box reductions. After that Paul Baecher et al. extended the original RTV framework in several respects using a more fine-grained and systematic way. The new framework provided by Paul Baecher et al. namely CAP framework clarifies the role of efficiency of adversaries and primitives within reductions and covers meta-reduction separations. Based on their enlightening work, we further consider the relations between different reductions such as the relation between the black-box use of efficient adversaries and non-black-box use of efficient adversaries. Then we prove the conjecture through a different approach.

Oracle separation methods are used in cryptography to rule out black-box reductions between cryptographic primitives. However, there is a big limitation for traditional oracle separation approach, namely it usually requires the number of adversaries is countable, and hence it cannot work in the non-uniform security model. To avoid the countability argument, Buldas and Niitsoo proposed an alternative oracle separation approach where the oracle extraction step is unnecessary. In this thesis we study the oracle separation approach so-called averaging approach, then we use the CAP framework to extend the notions of reductions in both the traditional oracle extraction based separation and the averaging-based separation.

Annotatsioon

Ligikaudu pool sajandit on musta kasti taandused olnud põhiline meetod, millega tõestada krüptograafilise konstruktsiooni turvalisust lähtudes primitiivide turvalisusest. Primitiivi $\mathcal{P}$ musta kasti taandus primitiivile $\mathcal{Q}$ on $\mathcal{P}$ konstruktsioon $\mathcal{Q}$ abil, mis ei kasuta $\mathcal{Q}$ siseehitust vaid ainult sisend-väljund käitumist. Kuid primitiive ei saa alati konstrueerida suvalisest teisest primitiivist. Ka selliseid taanduste olemasolule vastanduvaid negatiivseid tulemusi on palju kasutatud näitamaks teatud krüptograafiliste konstruktsioonide mitteeksisteerimist või ebaefektiivsust.

Reingold jt olid esimesed, kes pakkusid välja laialdaselt kasutatava raamistiku, nn RTV raamistiku, mis võimaldab klassifitseerida ja omavahel suhtestada erinevat tüüpi musta kasti taandusi. Peale neid on Paul Baecher jt laiendanud RTV raamistikku mitmes aspektis kasutades veelgi täpsemat ja süstemaatilisemat lähenemist. Uus, nn CAP raamistik selgitab täpsemini vastaste ja primitiivide efektiivsuse rolli taandustes ja katab ka nn meta-taandused ja eraldused. Käesolevas töös täpsustatakse CAP raamistikus mitmete klasside omavahelisi suhteid, nagu näiteks küsimust, kas efektiivsete vastaste musta kastina käsitlemine on ekvivalentne mitte-musta kasti käsitlusega. Selle hüpoteesi tõestamiseks kasutatakse erinevaid lähenemisi.

Oraakliga eraldamise meetodeid kasutatakse krüptograafias musta kasti taanduste mitteolemasolu tõestamiseks mingi kahe primitiivi vahel. Traditsiooniliste eraldusmeetodite kasutatavus on piiratud tänu neis kasutatavale eeldusele, et võimalike vastaste hulk on loenduv ja seetõttu ei välista traditsioonilised eraldusmeetodid näiteks mitte-ühtlaste taanduste olemasolu. Loenduvuse eelduse vältimiseks pakkusid Buldas, Laur ja Niitsoo välja alternatiivse lähenemise, nn. keskmistamise meetodi, kus ühe kindla eraldava oraakli valik ei ole vajalik. Selles töös uuritakse ka keskmistamise meetodi kohaldatavust CAP raamistikku.

Table of contents

List of Figures

List of Tables

1 Introduction

Constructing complex primitives from simpler ones is one of the most fundamental questions in cryptography. For nearly five decades, a lot of research has been done to show that existence of some important primitives implies the existence of other cryptographic primitives. It is well known that pseudo-random generators [11], statistically binding commitments [13], zero-knowledge proofs [14] and private-key encryption [15, 12, 11] were proved to be constructible from one-way functions. So far almost all security proofs for previous cryptographic constructions use a special kind of reductions called *black-box reductions*.

1.1 Black-Box Reductions

Informally, a black-box reduction of a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$ is a construction of $\mathcal{P}$ out of $\mathcal{Q}$ that does not use any internals of the primitive $\mathcal{Q}$ except the input and output behavior. It is, therefore, natural to study if all cryptographic primitives can be reduced to other cryptographic primitives through black-box reductions. Impagliazzo and Rudich [8] were the first to prove arguments against the existence of black-box reductions. They showed that constructions of key agreement (KA) based on one-way functions (OWP) imply a proof that $\mathsf{P} \neq \mathsf{NP}$. After that, many works had subsequently been done to address other questions on cryptographic constructions such as the relation between one-way functions and collision-resistant hash functions [10], one-way functions and one-way permutations [17], public-key encryption and trapdoor permutations [4], pseudorandom generators and one-way permutations [18]. By revisiting these negative results Reingold et al. [9] put forward a general framework about black-box reductions. They considered that there might be a weaker forms of black-box reductions between the primitives that do not contradict the existing separations. In their paper, they showed 7 types of different reductions and provided a widely adopted framework, called the *RTV framework*, to classify and relate different notions of black-box reductions. Inspired by the RTV framework, Paul Baecher et

al. [16] extended the framework in several respects by using a more fine-grained approach. They augmented more than 4 new different types of reductions to the RTV framework which formed a new framework called the $\text{CAP}_{(a,p)}$ framework. In [16] they distinguished explicitly between efficient and inefficient primitives, as well as adversaries for black-box reductions by using the index notation p and a. They also gave a comprehension of the relations of almost all reduction types in their framework. However, in their work, some relations between reduction types are still not clear. We will discuss these results in more detail in this thesis.

1.2 Black-Box Separations

A reduction is relativizing if it holds in the presence of any oracle, i.e., the reductions are valid in any computational model where ordinary Turing machines have access to a certain oracle. Hence, to show that there exist no black-box reductions from $\mathcal{P}$ to $\mathcal{Q}$, one has to find an oracle relativize to which there exists secure instance of $\mathcal{Q}$ but no instance of $\mathcal{P}$ is secure, which means there is no reduction from $\mathcal{P}$ to $\mathcal{Q}$ in the computational model with this oracle. This also shows that there is no black-box reduction from $\mathcal{P}$ to $\mathcal{Q}$ in the ordinary computational model, because such a reduction would also be valid relative to the oracle.

1.2.1 One-Oracle Techniques

Three main techniques have been used to show the black-box separations. The first technique was suggested by Impagliazzo and Rudich [8] to separate key agreement from one-way permutations. They first use a PSPACE-complete oracle to break any key agreement, then use a random permutation oracle to implement a one-way permutation. They combine these two oracles into one single oracle. In other words, this separation shows that there cannot exist a *relativizing reduction* from key agreement to one-way permutation. Relativizing separations is a commonly used technique for black-box separations [6, 8, 10, 6, 20].

1.2.2 Two-Oracles Techniques

The second technique for black-box separations is the so-called two-oracle technique which was first formalized by Hsiao and Reyzin [7]. The main difference between this technique and previous one is that we move the breaking oracle into the adversary such that the reduction can only access this oracle through the adversary. More formally, to show that there are no black-box constructions of primitive $\mathcal{P}$ from primitive $\mathcal{Q}$, it suffices to show that there exists an oracle Ω which is used to implement the primitive $\mathcal{Q}$ and an oracle Π which is used to break the primitive $\mathcal{P}$. For any oracle machines G, if G^{Ω} implements $\mathcal{P}$. then there exists some adversary A and for any algorithms R such that A^{Π} breaks G^{Ω} as $\mathcal{P}$, but no $R^{\Pi,\Omega}$ can break $\mathcal{Q}$. The two-oracles technique allows easier separations, as there is no need to combine these two oracles into one oracle as in [8]. The two-oracles technique has been applied successfully for many other researches such as [5, 7, 21, 22, 20]. In this thesis, we mainly use this type of technique to describe black-box reductions and separations.

1.2.3 Meta-reduction Techniques

The third technique for black-box separations is called meta-reductions. It was originally introduced by Boneh and Venkatenesan [19] and has gained significant attention recently. Roughly, a meta-reduction is a "reduction which can be used to prove the separation result", i.e., a meta-reduction $(\mathcal{P} \rightarrow \mathcal{Q})$ to $\mathcal{Q}$ is a proof that a reduction from $\mathcal{P}$ to $\mathcal{Q}$ exists only if there is no secure $\mathcal{Q}$. In a little more detail, the idea of meta-reductions can be described as follows. First, we assume that primitive $\mathcal{Q}$ exists and there also exists a black-box security reduction S from $\mathcal{P}$ to $\mathcal{Q}$. This means if A^{f} can break G^{f} as $\mathcal{P}$, then $S^{f,\mathsf{A}}$ can break the f as $\mathcal{Q}$. Now, for any such S, we construct a meta-reduction M that "simulates" the adversary A to the real reduction S. M can run S as it can simulate any answers that S queries to A. As long as S can not distinguish this simulated adversary from the real one M yields a procedure for breaking the primitive $\mathcal{Q}$ directly which is a contradiction. It turns out that meta-reductions can be classified according to black-box reductions. In this thesis, we will analyze the relations of different meta-reductions through the above notions for black-box reductions.

1.3 Non-uniform Reductions

In the non-uniform model, a poly-time oracle machine can be considered as a pair $(\mathsf{M}, \mathcal{A})$ where M is an ordinary poly-time oracle machine and $\mathcal{A} = \{a_k\}_{k\in\mathbb{N}}$ is an infinite sequence of (advice) bit-strings a_k with length ploynomial in k. For any oracle $\mathcal{O}$ and any input x, it is assumed that $\mathsf{M}^{\mathcal{O}}(x)$ has access to the advice string $a_{\{|x|\}}$. Usually, the advice strings are omitted for simplicity, but their presence must always be assumed when M is non-uniform. One of the most important facts is that since non-uniform Turing machines are infinite programs, there are uncountably many of them, whereas the set of ordinary Turing machines with finite programs is countable.

1.4 Oracle Extraction

All these separation techniques described above use separation oracles. In classical separation results of complexity theory, oracles are defined as fixed functions with a specific behavior. In cryptographic separations, it is very difficult to define a specific separation oracle. For example, it is hard to show that one-way functions exist relative to an oracle, because we do not know whether one-way functions exist in the standard computational model. So instead of defining a fixed oracle, we try to get the same result in an indirect way by using the so-called oracle extraction. Roughly speaking, we first need to define a certain probability distribution of oracles and then assume that the oracle is chosen randomly from that probability distributions. After that, we prove that the separation statements hold on average. Then we argue that there exists a particular choice of the oracle for which the statements also hold.

1.5 Averaging-Based Separation

However, there is a great limitation to use oracle extraction because it requires that the number of adversaries is countable. As we know, many practical primitives are required to be secure in the *non-uniform* security model, in which adversaries have advice strings, which means that the set of adversaries is not countable. Hence the oracle extraction cannot be used in the non-uniform security model. To avoid the countability argument, a different oracle separation approach, namely, *averaging approach* was suggested by

Buldas and Niitsoo [2]. To put it simply, they assumed that there exists a black-box reduction S which is universal for all f, then comparing the probabilistic separation condition and the average version of the reduction condition to get a contradiction. They proved that the averaging approach is capable of showing that there is no BNBa reductions (Definition. 14), or strong semi black-box reductions between two primitives in the non-uniform security model. In their paper, they also give an overview of four different types of reductions in both the traditional oracle extraction based separation and the averaging-based separation. In this paper, we try to extend their work in CAP framework and get the other three types of reductions in both the traditional oracle extraction-based separation and the averaging-based separation techniques

1.6 Outline and Contributions of this Thesis

The purpose of this thesis is two fold. First, we further consider the relations between different types of reductions, we get some new results for efficient adversaries in the CAP framework. Second, according to the work in [3] we complement all the notions of reductions for efficient adversaries in both the traditional oracle-extraction based separation and the averaging-based separation. We begin this thesis with some preliminary definitions in Chapter 2. Then in Chapter 3 we review the RTV and CAP framework for black-box reductions and define the notions of different reductions. In Chapter 3 and Chapter 4 we present our results:

In Chapter 3, we further consider some important questions about relations between different reductions such as whether black-box use of efficient adversaries is equivalent to non-black-box use. We use a different approach to prove an available result in [16], i.e., the equivalence of NNNa and NBNa reductions. Then we use this approach to prove the same equivalence result for BNNa and BBNa reductions. But when we try to use the same method to prove the equivalence of BBBa and BNBa reductions, we notice that there are different f and f' from a distribution $\mathcal{F}$ that S can break f with the access to f, but for f' not true. There is no contradiction result as previous reductions, hence it is probably not equivalent between BBBa and BNBa reductions.

In Chapter 4.2, We review the necessary fundamental about how to get a separation from the oracle-extraction, then we complement all notions of black-box reductions for efficient adversaries in the traditional oracle extraction-based separation. Finally, we give the proofs for each new separation condition.

In Chapter 4.3, We first study the necessary background about the averaging approach provide by Buldas and Niitsoo [2], then we complement all notions of black-box reductions for efficient adversaries in the averaging-based separation. Finally, we also give the proofs for each new separation condition.

2 Preliminaries

In this chapter, we set the basic notation used throughout the thesis.

In this paper, let $\mathbb{R}$ denote the set of all real numbers and $\mathbb{R}^+$ denote the set of all positive real numbers. We use $\mathbb{N}$ to denote the set of all natural numbers. We write $\{0,1\}^n$ to denote the set of binary strings of length n and $\{0,1\}^*$ to indicate the set of all finite binary strings. We use $|x|$ to denote the bit length of $x \in \{0,1\}^*$, i.e., $|x| = n$ if $x \in \{0,1\}^n$. For a distribution X, we write $x \leftarrow X$ for sampling that x is chosen randomly according to a distribution X. We write $x \leftarrow y$ for assigning value y to variable x. If X is a set, then by $x \leftarrow X$ we mean sampling from the uniform distribution on X. We use $\mathcal{F}$ to denote the set of all functions $\{0,1\}^* \to \{0,1\}^*$. We write $\cdot$ for a placeholder for function argument. For instance $f = G(\cdot,\cdot)$ means that f is a two argument function and $f(x,y) = G(x,y)$.

A set S is countable, iff there is an injection $\varphi : S \to \mathbb{N}$. We use $\mathcal{M}$ to denote the set of all ordinary Turing machines. The set of all ordinary Turing machines is countable, because we know that ordinary Turing machines have finite programs: (a) there is an injection $\iota : \mathcal{M} \to \{0,1\}^*$, and (b) there exists a bijection $\varphi : \{0,1\}^* \to \mathbb{N}$, and hence there is an injection $\varphi \circ \iota : \mathcal{M} \to \mathbb{N}$. According to the Cantor's diagonal argument, the set of non-uniform Turing machines $\mathcal{M}_n$ cannot be put into one-to-one correspondence with the set of natural numbers, hence the set of non-uniform Turing machines is not countable.

We use the Landau notation for describing asymptotic properties of functions. For functions $f, g : \mathbb{N} \to \mathbb{R}^+$, we write $f(k) = O(g(k))$ if $c \in \mathbb{N}$, and $f(k) \leq cg(k)$ for sufficiently large k. In particular, $f(k) = O(1)$ means that f is bounded. *Efficient computation* is modeled by a *poly-time* Turing machine M. A Turing machine M is poly-time, if for every x it runs in polynomial time $|x|^{O(1)}$. We use quantifiers $\underset{\text{pol}}{\forall} A$ to vary A over the set of all poly-time Turing machines, and $\underset{\text{pol}}{\exists} A$ means there exists a poly-time Turing machines A. We will often need to argue that an event occurs with very low probability. We write $f(k) = \omega(g(k))$ if $\lim_{k\to\infty} \frac{g(k)}{f(k)} = 0$. In particular, $f(k) = k^{-\omega(1)}$ indicates that $f(k)$ decreases faster than any polynomial, i.e., f is *negligible*.

Oracle Turing Machines: In this paper we will often talk about *oracle Turing machines*. An oracle Turing machine is an incompletely specified Turing machine M that is allowed to make oracle access to a function $\mathcal{O} : \{0,1\}^* \to \{0,1\}^*$. Whenever M makes a query x to $\mathcal{O}$, it receives the answer $\mathcal{O}(x)$ in a single computation step. In this case, we write $\mathsf{M}^{\mathcal{O}}$ to indicate a machine M with oracle access to $\mathcal{O}$. Here oracle access means that machine M does not use any internals of the function $\mathcal{O}$ except the input and output behavior. In particular, the Turing machine M does not learn how the computation of $\mathcal{O}$ is actually performed. Note that the function $y \leftarrow \mathcal{O}(x)$ is not necessary to be computable, but it still has a conditional running time $t(|x|)$, which means that we do not care about actual computations "inside" the $\mathcal{O}$. The running time of $\mathsf{M}^{\mathcal{O}}$ is determined by the conditional running time of oracle calls—each call $\mathcal{O}(x)$ takes $t(|x|)$ steps, where $|x|$ denotes the bit-length of x. We say that M is a *poly-time oracle machine* if $\mathsf{M}^{\mathcal{O}}$ runs in poly-time, whenever $\mathcal{O}$ is poly-time. i.e., if $t(|x|) = |x|^{O(1)}$.

A *non-uniform* poly-time oracle machine is a pair $(\mathsf{M}, \mathcal{A})$ where M is an ordinary poly-time oracle machine and $\mathcal{A} = \{a_k\}_{k\in\mathbb{N}}$ is an infinite sequence of (advice) bit-strings a_k with length $k^{O(1)}$. For any oracle $\mathcal{O}$ and any input x, it is assumed that $\mathsf{M}^{\mathcal{O}}(x)$ has access to the advice string $a_{|x|}$. Usually, the advice strings are omitted for simplicity, but their presence must always be assumed when M is non-uniform. One of the most important facts is that since non-uniform Turing machines are with infinite families of advice strings, the set of all non-uniform (oracle) Turing machines is uncountable, whereas the set of ordinary Turing machines is countable.

By an *oracle function* we mean a family of functions φ_k with domain of the set $\mathfrak{S}$ of all oracles and codomain of all finite binary strings $\{0,1\}^*$, i.e., it is $\varphi_k : \mathfrak{S} \to \{0,1\}^*$. Note that oracle functions may give non-trivial information about oracles that cannot be efficiently collected with oracle calls. For instance, there exists an oracle function φ such that $\varphi_k(\mathcal{O}) = 1$, if $\exists x \in \{0,1\}^k, \mathcal{O}(x)$ is odd, and otherwise $\varphi_k(\mathcal{O}) = 0$. The quantifier $\underset{\text{of}}{\forall}\varphi$ means that the quantified φ varies over all oracle functions, and $\underset{\text{of}}{\exists}\varphi$ means that there exists an oracle function φ.

A probability space is a triple $(\Omega, \mathcal{F}, \Pr)$ consisting of : (1) The sample space Ω — an arbitrary non-empty set. (2) The σ-algebra $\mathcal{F} \subseteq 2^\Omega$ — a set of subsets of Ω, called events, such that: $\mathcal{F}$ contains the sample space: $\Omega \in \mathcal{F}$. $\mathcal{F}$ is closed under complements: if $A \in \mathcal{F}$, then also $(\Omega \backslash A) \in \mathcal{F}$. $\mathcal{F}$ is closed under countable unions: if $A_i \in \mathcal{F}$ for $i = 1, 2, \ldots$, then also $(\bigcup_{i=1}^{\infty} A_i) \in \mathcal{F}$. (3) The probability measure $\Pr : \mathcal{F} \to [0,1]$ — a function on $\mathcal{F}$ such that: $\Pr$ is countably additive: if $\{A_i\}_{i=1}^{\infty} \subseteq \mathcal{F}$ is a countable collection of pairwise disjoint sets, then $\Pr(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} \Pr(A_i)$. The measure of

entire sample space is equal to one: $\Pr(\Omega) = 1$.

A random variable $X : \Omega \to E$ is a measurable function from the set of possible outcomes Ω to some set E. We usually require Ω to be a probability space and E to be the set of all real numbers $\mathbb{R}$. Let X be a random variable taking values $x_1, x_2, \cdots$ with probabilities $p_1, p_2, \cdots$ respectively. The expected value of this random variable is the infinite sum $\mathsf{E}[X] = \sum_{i=1}^{\infty} x_i p_i$.

If $\mathcal{D}$ is a distribution and $G(x)$ is a predicate, $\Pr_{x \leftarrow \mathcal{D}}[G(x)]$ denotes the probability that $G(x)$ is true after the assignment of x from the distribution $\mathcal{D}$. Similarly, if $\mathcal{D}$ is a distribution and $f(x)$ is a predicate, $\mathbf{E}_{x \leftarrow \mathcal{D}}[f(x)]$ denotes the expected value of $f(x)$ after the assignment of x from the distribution $\mathcal{D}$.

2.1 Basic Lemmas

Here we provide several lemmas which we will use in this thesis. The following statements of lemmas are due to [3].

Lemma 1 (Borel-Cantelli) *Suppose that* $\{\mathsf{E}_n : n \geq 1\}$ *is a sequence of events in a probability space. Let* $\mathsf{E}_\infty = [\mathsf{E}_n \text{ occurs for infinitely many } n] = \bigcap_{k=1}^{\infty} \bigcup_{n=k}^{\infty} \mathsf{E}_n$. *If* $\sum_n \Pr[\mathsf{E}_n] < \infty$ *then* $\Pr[\mathsf{E}_\infty] = 0$*, with probability zero only a finite number of the events occur.*

Proof. Indeed, let $B_k = \bigcup_{n=k}^{\infty} \mathsf{E}_k$. If $x \in \mathsf{E}_\infty$ then $x \in \cap_k B_k$, because otherwise x only belongs to a finite sequence $\mathsf{E}_1, \ldots, \mathsf{E}_{k-1}$ of events. Hence, $\mathsf{E}_\infty \subseteq \cap_k \mathsf{B}_k$ and $\Pr[\mathsf{E}_\infty] \leq \Pr[\cap_k \mathsf{B}_k] \leq \Pr[\mathsf{B}_k]$. From $\sum_n \Pr[\mathsf{E}_n] < \infty$ it follows that for every $\epsilon > 0$ there is k such that $\sum_{n=k}^{\infty} \Pr[\mathsf{E}_k] < \epsilon$. Thus, $\Pr[\mathsf{E}_\infty] \leq \Pr[\mathsf{B}_k] = \Pr[\cup_{n=k}^{\infty} \mathsf{E}_k] \leq \sum_{n=k}^{\infty} \Pr[\mathsf{E}_k] < \epsilon$, which implies $\Pr[\mathsf{E}_\infty] = 0$. □

Lemma 2 (Markov's Inequality) *For any* $h > 0$, $\Pr[|X| \geq h] \leq \frac{\mathsf{E}[|X|]}{h}$. *When* X *only takes non-negative values then for any* $h > 0$ $\Pr[X \geq h] \leq \frac{\mathsf{E}[X]}{h}$, *where* $\mathsf{E}[X]$ *denotes the expectation of X.*

Lemma 3 (Jensen's Inequality) *if* $f(x)$ *is a convex function and* $X \in \{x_i : 1, ..., N\}$ *is a random variable with probabilities* $\Pr[x_i]$ *where* $\sum_{i=1}^{N} \Pr[x_i] = 1$, *then* $f(\mathsf{E}\{X\}) \leq \mathsf{E}\{f(X)\}$ *or* $f(\sum_{i=1}^{N} x_i \Pr[x_i]) \leq \sum_{i=1}^{N} x_i f(i_x) \Pr[x_i]$.

Lemma 4 (Probabilistic Argument) *Let $\mathcal{F}$ be a probability space and G be a predicate function. Then* $\Pr_{f \leftarrow \mathcal{F}}[G(f)] > 0 \;\Rightarrow\; \exists f\colon\, G(f)$.

Lemma 5 (Countability Argument) *Let $\mathcal{F}$ be a probability space and $G(f, A)$ be a predicate function where A varies over all poly-time Turing machines, then*

$$\underset{pol}{\forall} A\colon\, \Pr_{f \leftarrow \mathcal{F}}[G(f, A)] = 1 \quad \Rightarrow \quad \Pr_{f \leftarrow \mathcal{F}}\left[\underset{pol}{\forall} A\colon\, G(f, A)\right] = 1 \ .$$

Proof. Countable intersection of measure one sets is a measure one set. □

Lemma 6 (Negligible Average Argument) *Let $\mathcal{F}$ be a distribution so that for every $f \leftarrow \mathcal{F}$ there is a real-valued function $\delta_f\colon \mathbb{N} \to [0,1]$. If $\mathop{\mathbf{E}}_{f \leftarrow \mathcal{F}}[\delta_f(k)] = \varepsilon(k) = k^{-\omega(1)}$, then $\delta_f(k) = k^{-\omega(1)}$ for measure one of f's.*

Proof. As $\Pr_{f \leftarrow \mathcal{F}}[\delta_f(k) > k^2 \cdot \varepsilon(k)] \;\leq\; k^{-2}$ and $\Pr_{f \leftarrow \mathcal{F}}[\delta_f(k) \leq k^2 \cdot \varepsilon(k)] \;\geq\; 1 - k^{-2}$ by Markov inequality, we define E_k as the event that $\delta_f(k) > k^2 \cdot \varepsilon(k)$. Now we use the Borel-Cantelli lemma and $\sum_k \Pr[\mathsf{E}_k] \leq \sum_k k^{-2} < \infty$ to imply

$$\Pr_{f \leftarrow \mathcal{F}}[\text{"}\delta_f(k) > k^2 \cdot \varepsilon(k) \text{ for infinitely many } k\text{-s"}] = \Pr[\mathsf{E}_\infty] = 0 \ .$$

Thus, for measure one of f's: $\exists k_0 \forall k > k_0 : \delta_f(k) \leq k^2 \cdot \varepsilon(k) = k^{-\omega(1)}$. □

Lemma 7 (Overwhelming Average Argument) *Let $\mathcal{F}$ be a distribution so that for every $f \leftarrow \mathcal{F}$ there is a function $\delta_f\colon \mathbb{N} \to [0,1]$. If $\mathop{\mathbf{E}}_{f \leftarrow \mathcal{F}}[\delta_f(k)] = 1 - k^{-\omega(1)}$, then $\delta_f(k) = 1 - k^{-\omega(1)}$ for measure one of f's.*

Proof. $\mathop{\mathbf{E}}_{f \leftarrow \mathcal{F}}[1 - \delta_f(k)] = 1 - \mathop{\mathbf{E}}_{f \leftarrow \mathcal{F}}[\delta_f(k)] = 1 - (1 - k^{-\omega(1)}) = k^{-\omega(1)}$, which by Lemma 6 implies that $1 - \delta_f(k) = k^{-\omega(1)}$ for measure one of f's. □

Lemma 8 *There exist quantities $\delta_i(k) = k^{-\omega(1)}$ for which $\mathop{\mathbf{E}}_{i}[\delta_i(k)] \neq k^{-\omega(1)}$.*

Proof. Let $I = \{1, 2, \ldots\}$ and $p_i = \frac{6}{\pi^2 i^2}$ for all $i \in I$. Then $\sum_{i \in I} p_i = 1$. For all $i \in I$ we define the function δ_i by $\delta_i(k) = \delta_{ik}$, where δ_{ik} is the Kronecker delta. Now we define a probability space on $\{\delta_i\}_{i \in I}$ such that $\mathsf{Pr}[\delta_i] = p_i$ for all $i \in I$. Note that $\delta_i(k) = k^{-\omega(1)}$ for all $i \in I$ but the average of all δ_i-s is non-negligible, because $\mathop{\mathbf{E}}_{i}[\delta_i(k)] = \frac{6}{\pi^2} \cdot k^{-2} = k^{-O(1)} \neq k^{-\omega(1)}$. □

2.2 Cryptographic Primitives and Security

Cryptographic primitives can be considered as the most basic building blocks when we create complex cryptographic constructions. It is natural that an instance of a primitive can be represented as a function $f\colon \{0,1\}^* \to \{0,1\}^*$. For example, in the case of some common primitives such as one-way permutations or collision-resistant hash functions, f is simply the one-way function or hash function itself. In some more complicated cases such as encryption schemes, it may make more sense to define a primitive as three different functions, including the key generation functions, the encryption functions, and the decryption functions. However, we can usually concatenate these functions into one single function – we just use the first few bits of the argument to divide the function into the sub-functions. This means that we can still formalize the primitive as one single function.

An instance f of a primitive $\mathcal{P}$ need to meet various *structural* and *correctness* requirements. For example, we require that the decryption of an encryption of a plaintext m will recover m for encryption schemes. Formally, we would define a correctness check predicate function $\mathcal{C}$, in which $\mathcal{C}(f) = 1$, if $f \in \mathcal{P}$, and $\mathcal{C}(f) = 0$, if $f \notin \mathcal{P}$. The function $\mathcal{C}$ just helps to check the syntax of the instance and does not say anything about the security of f. For example, every permutation can be an instance of the one-way permutation primitive, but not necessarily a secure instance.

Therefore, we also need to define the *security* of primitives. In this work, instead of using the definition provided by Reingold et al., we use a more specific (but still sufficiently general) definition of security given in [2], where the breakage advantage is a real-valued function that also depends on the *security parameter* k which is usually tied to the actual input(or output) lengths of the primitive. Formally, primitives have an *advantage function* $\mathrm{ADV}_k^{\mathcal{P}}(\cdot,\cdot)$, which given as input the security parameter $k \in \mathbb{N}$, an instance f of $\mathcal{P}$, and an oracle Turing machine A (an *adversary*) returns a real number $\mathrm{ADV}_k^{\mathcal{P}}(\mathsf{A}, f) \in [0,1]$. We say that A *breaks* an instance f of $\mathcal{P}$ if $\mathrm{ADV}_k^{\mathcal{P}}(\mathsf{A}, f) \neq k^{-\omega(1)}$. If no poly-time Turing machine A breaks f then f is said to be *secure*. The whole definition of primitives, adversaries and advantage with oracles are as follows.

Definition 1 *A cryptographic* primitive *is a pair* $(\mathcal{P}, \mathrm{ADV}_k^{\mathcal{P}}(\cdot,\cdot))$ *where* $\mathcal{P} \subseteq \mathcal{F}$ *and* $\mathrm{ADV}_k^{\mathcal{P}}(\cdot,\cdot) : \mathcal{F} \times \mathcal{F} \to [0,1]$.

Definition 2 *Let $\mathcal{P}$ be the set of all functions $f \in \mathcal{F}$ computable in poly-time. If $\mathcal{O}$ is an oracle, then by $\mathcal{P}^{\mathcal{O}}$ we mean that the set of all functions $f \in \mathcal{F}$ computable by poly-time oracle machines with the oracle $\mathcal{O}$.*

Definition 3 *Let $M^{\mathcal{O}}_{\varphi(\mathcal{O})}$ be a non-uniform oracle Turing Machine with a sequence of advice bit-strings $\mathcal{A} = \varphi(\mathcal{O}) = \{\varphi_0(\mathcal{O}).\varphi_1(\mathcal{O}), \cdots \varphi_k(\mathcal{O})\}$ with length $k^{O(1)}$. $\mathcal{P}^{\mathcal{O}}_{\varphi(\mathcal{O})}$ is the set of the poly-time non-uniform oracle Turing Machine with advice string $\varphi(\mathcal{O})$ that have access to the oracle $\mathcal{O}$.*

Definition 4 *We say that $\mathsf{A} \in \mathcal{F}$ breaks $f \in \mathcal{P}$, if $\mathrm{ADV}^{\mathcal{P}}_k(\mathsf{A}, f) \neq k^{-\omega(1)}$. Let $\mathfrak{S} \subseteq \mathcal{F}$, $f \in \mathcal{P}$ is $\mathfrak{S}$-secure, if $\forall \mathsf{A} \in \mathfrak{S}, \mathrm{ADV}^{\mathcal{P}}_k(\mathsf{A}, f) = k^{-\omega(1)}$. Similarly, an instance f of $\mathcal{P}$ is secure relative to an oracle $\mathcal{O}$, if f is $\mathcal{P}^{\mathcal{O}}$-secure. An instance f of $\mathcal{P}$ is secure relative to a φ-leaky oracle $\mathcal{O}$, if f is $\mathcal{P}^{\mathcal{O}}_{\varphi(\mathcal{O})}$-secure.*

Definition 5 *We say that a primitive $\mathcal{P}$ exists if there is an efficient implementation $f \in \mathcal{P}$ that is $\mathcal{P}$-secure.*

Here we want to argue about that our definitions do not say anything about the efficiency of instance f. The function may even be non-computable, as long as the advantage that can be gained by any adversary is negligible. In practice, one needs an instantiation of a primitive that is both efficient and secure, and an *efficient* implementation of $\mathcal{P}$ is an implementation of $\mathcal{P}$ which is computable by poly-time Turing machines.

3 Notions of Cryptographic Reductions

In this chapter, we first briefly introduce the concept of reductions from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$. Then in Section 3.1, we review the RTV framework [9] and give the definitions of several different types of black-box reductions. Based on that we summarize the basic relations between different reductions in RTV framework. In Section 3.2, we compare the RTV and CAP framework [16] and give the whole picture of the CAP framework. In Section 3.3, we consider the potential relations between the reductions based on the existing results in [16] and prove a conjecture proposed by Reingold et al.[9]. In Section 3.4, we briefly introduce the concept of parametrized black-box reductions. Finally in Section 3.4, we briefly introduce the poly-preserving reduction and make a comparison between the poly-preserving reductions and general reduction.

In cryptography, a reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$ usually means that (1) $\mathcal{P}$ can be efficiently implemented by $\mathcal{Q}$. (2) If there is an adversary A that breaks $\mathcal{P}$, then there is an adversary A' that breaks $\mathcal{Q}$. In simple words, it means that either $\mathcal{P}$ exists or $\mathcal{Q}$ does not exist. Most common cryptographic reductions are black-box reductions, i.e., the construction $\mathcal{P}$ and adversary A' use $\mathcal{Q}$ and A as black boxes. To compare different black-box reductions, we start to review the framework of black-box reductions.

3.1 The RTV Framework for Black-box Reductions

As we mentioned the first widely adopted framework to classify black-box reductions was called RTV framework and provided by Reingold, Trevisan and Vadhan [9]. There are 7 different types of reductions in this framework which we define as Definition 6-12.

Definition 6 (Fully black-box reduction) *There exists a fully-BB reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, if there exist two poly-time oracle machines G and S such that:*

(C) *If f implements $\mathcal{Q}$ then G^f implements $\mathcal{P}$.*

(S) *For every instance $f \in \mathcal{Q}$, if A breaks G^f (as $\mathcal{P}$) then $S^{\mathsf{A},f}$ breaks f (as $\mathcal{Q}$).*

As a consequence, the existence of a secure $\mathcal{Q}$ implies the existence of a secure $\mathcal{P}$. For this type of black-box reductions, we have the most strong restriction that the construction G and the reduction algorithm S must treat f in a black-box way, meanwhile, the reduction algorithm S also has to treat the adversary A in a black-box way. The next, less restricted, notion is *semi-black-box* reduction where S may depend on the adversary A and an instance f.

Definition 7 (Semi black-box reduction) *There exists a semi-black-box reduction from a primitive $\mathcal{P}$ to to a primitive $\mathcal{Q}$, if there exists a poly-time oracle machine G such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For every instance $f \in \mathcal{Q}$ and a poly-time oracle machine A, there exists a poly-time oracle machine S such that if A^f breaks G^f then S^f breaks f.*

Based on semi black-box reductions we add additional restriction to adversary A that it cannot get oracle access to f then we get the definition of a *weakly-black-box* reduction.

Definition 8 (Weakly black-box reduction) *There exists a weakly black-box reduction from a primitive $\mathcal{P}$ to to a primitive $\mathcal{Q}$, if there exists a poly-time oracle machine G such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For every instance $f \in \mathcal{Q}$ and a poly-time machine A, there exists a poly-time oracle machine S such that if A breaks G^f then S^f breaks f.*

The difference between the two reduction types is very subtle. Essentially, weakly-bb does not change the fact that reduction algorithm S treats the adversary A and instance f in non-black-box way, so we consider they are different branches of the same type of black-box reductions. The next reductions are *relativizing reductions* which can be described as follows.

Definition 9 (Relativizing reduction) *There exists a relativizing reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, if for all oracles Π, the primitive $\mathcal{P}$ exists relative to Π whenever $\mathcal{Q}$ exists relative to Π.*

In general, relativizing reduction and semi-BB reduction are equivalent, if it is possible to "embed" an arbitrary oracle into $\mathcal{Q}$ as the proof in [9]. Finally, we consider two additional notions of reductions that derive from semi-BB and weakly-BB. As we mentioned they can be considered as different branches of the same type of black-box reductions where the reduction algorithm S depends on A and f, simultaneously construction G must be universal for all valid instances $f \in \mathcal{Q}$ and as such, specific properties of an instance f cannot be used in the construction. Now we allow the construction G do depend on f, we get the definition of $\forall\exists$ *semi-BB* and $\forall\exists$ *weakly-BB* reductions.

Definition 10 ($\forall\exists$ Semi BB-reduction) *There is a $\forall\exists$ semi-BB reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff for any correct implementation f of $\mathcal{Q}$:*

(C) *There exists a poly-time oracle machine G^f that correctly implements $\mathcal{P}$;*

(S) *For every instance $f \in \mathcal{Q}$ and for any poly-time oracle machines A, there exists a poly-time oracle machine S such that if A^f breaks G^f, then S^f breaks f.*

Definition 11 ($\forall\exists$ Weakly BB-reduction) *There is a $\forall\exists$ weakly BB-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff for any correct implementation f of $\mathcal{Q}$:*

(C) *There exists a poly-time oracle machine G^f that correctly implements $\mathcal{P}$;*

(S) *For every instance $f \in \mathcal{Q}$ and for any poly-time machines A, there exists a poly-time oracle machine S such that if A breaks G^f, then S^f breaks f.*

Definition 12 (Free Reduction) *There exists a free reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, if $\mathcal{P}$ exists whenever $\mathcal{Q}$ exists.*

Note that we do not give the reduction S with oracle access to the poly-time adversary A for all cases where S depends on A in a non-black box way. In these cases, a poly-time reduction S can completely simulate the efficient adversarial algorithm A.

The RTV framework is a partial order hierarchy with fully black-box reductions being the strongest type of reductions and $\forall\exists$ weakly BB-reductions being the weakest. Existence of a reduction of a stronger type trivially implies the existence of reductions of all weaker types. Figure 3.1 shows the relations between these classes. Note that Reingold et al. [9] pointed out that weakly-BB are as powerful as free reductions and black-box separations result are presumably impossible in the weakly level and for free reductions. Hence, in this paper we only capture the reductions above the semi level.

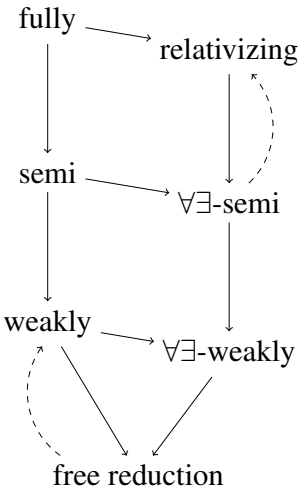

Figure 3.1: Relations between notions of reductions in the original RTV framework. Arrows go from more restricted forms of reductions to less restricted ones. Dashed arrows indicate that the equivalence relation exists in some interesting cases.

3.2 The CAP Framework for Black-box Reductions

The CAP framework is the latest classification method on both black-box constructions and separations. Here let us first make a brief comparison between the CAP and the RTV frameworks. In the RTV framework we mainly consider three conditions when we give the notions for reductions: (1) Whether the construction treats the primitive in a black-box way or not. (2) Whether the reduction algorithm treat the primitive and the adversary together in a black-box way or not. (3) Whether the adversary can get oracle access to the primitive or not. As we mentioned the third condition actually does not impact the fact of type for black-box reductions. So, from each combination of different cases, we can theoretically get 4 notions of reductions except the relativizing and the free reductions in the RTV framework. The CAP framework augments the basic notions of original RTV framework in various directions. And most of all, they further subdivide the second condition into two cases: (1) Whether the reduction algorithm treats the adversary in a black-box or non-black-box way. (2) Whether the reduction algorithm treats the primitive in a black-box or non-black-box way. Based on that they introduce a more descriptive three-character "CAP" notation with words from the language $\{\text{B,N}\}^3$ to indicate whether the construction (C), the adversary in the reduction (A), or the primitive in the reduction (P) is treated in a black-box (B) or non-black-box (N) way. For example, a *BBB-reduction* in CAP notation means the construction, adversary and primitive are all treated in a black-box way which is equivalent to a fully black-box reduction in the RTV framework. Similarly, a *BNN-reduction* in the CAP framework is

equivalent to a semi-black-box reduction in the RTV framework, in which reduction can depend on the description of the adversary and the primitive, and only the construction is black box. A *NNN-reduction* in the CAP framework is equivalent to a $\forall\exists$ semi black-box reduction in the RTV framework, in which none of construction, adversary or primitive are black-box. Finally a $\forall\exists$ fully black-box reduction (This type of reduction actually was not formally defined in [9], the order of the quantifiers can be found in Table 3.2) is equivalent to a *NBB reduction* in the CAP framework in which adversary and primitive are black-box, and only the construction can depend on the code of primitive. The Table 3.1 show the relation between CAP framework and RTV framework.

CAP	RTV
BBB	Fully-BB
BNB	
BBN	
BNN	Semi-BB
NBB	$\forall\exists$ Fully-BB
NBN	
NNB	No meaning
NNN	$\forall\exists$ Semi-BB

Table 3.1: Corresponding relation between CAP framework and RTV framework. Strictly, in addition to BBB-reduction, the equivalence relations only exist in CAP for efficient adversaries.

If we do not consider the weakly dimension of the RTV framework, it is easy to see that the RTV framework only covers half of the all 8 possibilities for the CAP framework. Note that in CAP there are only 7 reasonable notions of all 8. We need except the NNB reduction, because from the restricted of the NNB reduction, we know that the construction may depend on the primitive, the reduction algorithm may depend on the adversary, and the reduction should be universal for the primitive. Thus, the order of the quantifiers is $(\forall A\exists S\forall f\exists G)$ in which the construction can now depend on the adversary. In this sense, it has no meaning in cryptology.

Now let us fill in the missing types in RTV framework. We first start from the notions of *BBN-reduction* in which the construction makes black-box calls to the primitive, the reduction has to work for all adversaries, but may depend on the primitives. The second one is *BNB-reduction* in which the construction also makes black-box calls to the primitive, the reduction is universal for all primitives but may depend on adversary. The last one is *NBN-reduction* in which the construction makes non-black-box use of the primitive, the reduction has to work for all adversaries, but may depend on the primitive.

Strict definitions of each are as follows.

Definition 13 (BBN-reduction) *There exists a BBN-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff there exists a poly-time oracle machine G such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For every instance $f \in \mathcal{Q}$, there exists a poly-time oracle machine S for all machines A such that if A^f breaks G^f then $S^{f,A}$ breaks f.*

Definition 14 (BNB-reduction) *There exists a BNB-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff there exists a poly-time oracle machine G such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For all machines A there exists a poly-time oracle machine S such that for every instance $f \in \mathcal{Q}$ if A^f breaks G^f then $S^{f,A}$ breaks f.*

Definition 15 (NBN-reduction) *There is a NBN-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff for any correct implementation f of $\mathcal{Q}$:*

(C) *There exists a poly-time oracle machine G^f that correctly implements $\mathcal{P}$;*

(S) *For every instance $f \in \mathcal{Q}$, there exists a poly-time oracle machine S such for any machines A, if A^f breaks G^f, then $S^{f,A}$ breaks f.*

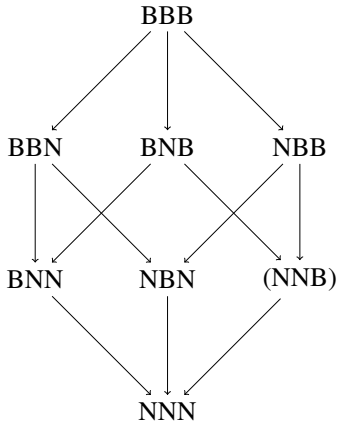

Figure 3.2: Relations between the notions of reductions in the CAP framework. Arrows go from more restricted forms of reductions to less restricted ones. The NNB reduction can be ignored.

We can view the complete picture of the CAP framework in Figure 3.2. Note that the CAP framework is also a partial order hierarchy and the basic relations of the RTV framework still apply to the CAP framework that existence of a strong reduction type can trivially imply the existence of reductions of all weaker types.

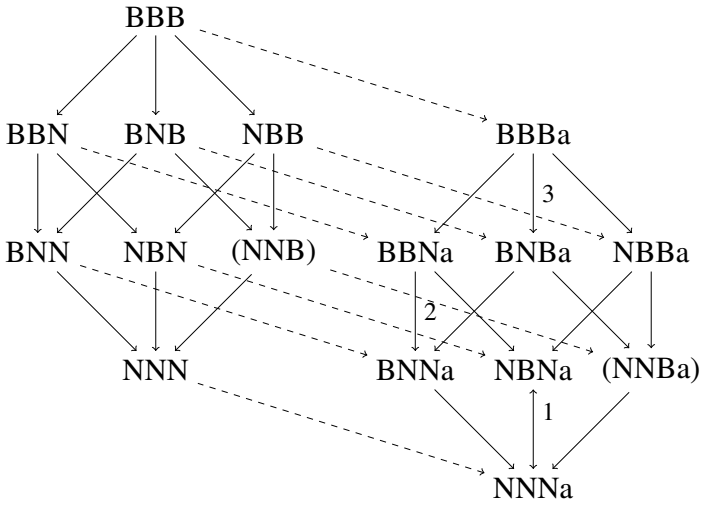

Figure 3.3: The CAP framework for (in)efficient adversaries. Arrows go from more restricted forms of reductions to less restricted ones. The dashed arrows designate implications

The CAP framework also provides a further classification for black-box reduction based on the (in)efficiency of the primitives and adversaries as show in Figure 3.3. These cases are similar to the weakly level in the RTV framework, and can be considered as same type of black-box reduction in different dimension. Not that previous corresponding relation between the RTV and CAP framework only work for efficient adversaries. In the CAP framework the suffix 'a' denote an efficiency requirement on the adversary, the suffix 'p' indicate that primitives are efficiently computable, hence a *BBNa-reduction* means that the construction G is universal for all primitives, the reduction S is universal for all poly-time oracle machines A, and may depend on primitive.

Definition 16 (BBNa-reduction) *There exists a BBNa-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, if there exists a poly-time oracle machine G such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For every instance $f \in \mathcal{Q}$, there exists a poly-time oracle machine S for all poly-time oracle machine A such that if A^f breaks G^f then $S^{f,A}$ breaks f.*

We know that the differences between definitions with efficient adversaries and with

inefficient adversaries are very small, and nearly half of the definitions for efficient adversaries were already given in the RTV framework. Table 3.2 summarizes all the notions. The complete definitions for the remaining types of reductions can be found as follows.

CAP	Quantification				RTV
BBBa	$\underset{\text{pol}}{\exists} G$	$\underset{\text{pol}}{\exists} S$	$\forall f$	$\underset{\text{pol}}{\forall} A$	
BNBa	$\underset{\text{pol}}{\exists} G$	$\underset{\text{pol}}{\forall} A$	$\underset{\text{pol}}{\exists} S$	$\forall f$	
BBNa	$\underset{\text{pol}}{\exists} G$	$\forall f$	$\underset{\text{pol}}{\exists} S$	$\underset{\text{pol}}{\forall} A$	
BNNa	$\underset{\text{pol}}{\exists} G$	$\underset{\text{pol}}{\forall} A$	$\forall f$	$\underset{\text{pol}}{\exists} S$	Semi-BB
NBBa	$\underset{\text{pol}}{\exists} S$	$\forall f$	$\underset{\text{pol}}{\exists} G$	$\underset{\text{pol}}{\forall} A$	$\forall\exists$ Fully-BB
NBNa	$\forall f$	$\underset{\text{pol}}{\exists} G$	$\underset{\text{pol}}{\exists} S$	$\underset{\text{pol}}{\forall} A$	
NNNa	$\forall f$	$\underset{\text{pol}}{\exists} G$	$\underset{\text{pol}}{\forall} A$	$\underset{\text{pol}}{\exists} S$	$\forall\exists$ Semi-BB

Table 3.2: The CAP framework for efficient adversaries

Definition 17 (BBBa-reduction) *There exists a BBBa-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, if there exist two poly-time oracle machines G and S such that:*

(C) *If f implements $\mathcal{Q}$ then G^f implements $\mathcal{P}$.*

(S) *For every instance $f \in \mathcal{Q}$ and all poly-time oracle machines A, if A^f breaks G^f, then $S^{f,A}$ breaks f.*

Definition 18 (BNBa-reduction) *There exists a BNBa-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff there exists a poly-time oracle machine G such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For all poly-time oracle machines A there exists a poly-time oracle machine S such that for every instance $f \in \mathcal{Q}$ if A^f breaks G^f then S^f breaks f.*

Definition 19 (NBBa-reduction) *There is a NBBa-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff there exists a poly-time oracle machine S such that:*

(C) *If f correctly implements $\mathcal{Q}$ then G^f correctly implements $\mathcal{P}$.*

(S) *For any poly-time oracle machine A such that if A^f breaks G^f, then $S^{f,A}$ breaks f.*

Definition 20 (NBNa-reduction) *There is a NBNa-reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, iff for any correct implementation f of $\mathcal{Q}$:*

(C) *There exists a poly-time oracle machine G^f that correctly implements $\mathcal{P}$;*

(S) *For every instance $f \in \mathcal{Q}$, there exists a poly-time oracle machine S such for any poly-time machines A, if A^f breaks G^f, then $S^{f,A}$ breaks f.*

3.3 Relations of Black-box Reductions in the CAP

After defining the notions of black-box reductions in the CAP framework, the next question is how these definitions are related. As I mentioned, most of the results for this question had already been done in [16]. The Figure 3.3 shows the existing relations between reductions in the CAP framework. However, we notice that there still exists some questions which have not been solved yet. For instance, the conjecture about whether black-box use of efficient adversaries is equivalent to non-black-box use. This conjecture was first proposed by Reingold et al. [9]. They guess that there is no inherent restriction in treating the adversary as a black-box. From the Figure 3.3 it is easy to see that if we want to confirm this conjecture, we need to prove arrows 1, 2, 3 are reversible. Paul Baecher et al. [16] proved that the arrow 1 is reversible which means that NNNa and NBNa are equivalent. Here we use a different approach to prove the same result.

Theorem 1 (Equivalence of NNNa and NBNa) *For all primitives $\mathcal{P}$ and $\mathcal{Q}$, there is a NBNa-reduction for efficient adversaries A, if and only if there is a NNNa-reduction.*

Proof. Existence of a reduction of a strong type trivially implies the existence of reductions of all weaker types. Using straightforward logical deductions, it follows that NBNa-reductions imply NNNa-reductions. Now we want to prove the converse. First we assume that there is a NNNa-reduction (the order of the quantifiers is $\forall f \exists G \forall A \exists S$) between the primitives $\mathcal{P}$ and $\mathcal{Q}$, but there is no NBNa-reduction. From the Table-3.2 we know that it means that there exists an $f \in \mathcal{Q}$ such that for any poly-time oracle machines G and for all poly-time oracle machines S, there exists an efficient adversary A (the order of the quantifiers is $\exists f \forall G \forall S \exists A$) such that $[\mathsf{A}^f \mathbf{br} G^f \wedge S^{\mathsf{A},f} \not\mathbf{br} f]$. From the assumption we can pick a fixed f then pick a G which depends on f such that:

$$(\mathrm{I}) \underset{\text{pol}}{\forall} \mathsf{A} \underset{\text{pol}}{\exists} S\colon [\mathsf{A}^f \mathbf{br} G^f \Rightarrow S^f \mathbf{br} f] \ .$$

$$(\mathrm{II}) \underset{\text{pol}}{\forall} S \underset{\text{pol}}{\exists} \mathsf{A}\colon [\mathsf{A}^f \mathbf{br} G^f \wedge S^{\mathsf{A},f} \not\mathbf{br} f] \ .$$

(1) (II)$\Longrightarrow$(II′) $\underset{\text{pol}}{\exists}\,\mathsf{A}\,:\,[\mathsf{A}^f\ \mathbf{br}\ G^f]$. Because, if $\underset{\text{pol}}{\forall}\,\mathsf{A}\,:\,[\mathsf{A}^f\ \not\mathbf{br} G^f]$, then by choosing any poly-time oracle machines in (II), there exits at least one S_0, for example the outputs of S_0 are always 0. Let A in (II′), apply (I) $[\mathsf{A}^f\ \mathbf{br}\ G^f]$, then there exists a S' which depends on A satisfies:

$$\underset{\text{pol}}{\exists}\,S'\colon [S'^f\ \mathbf{br}\ f]\ . \tag{3.1}$$

For case (II) we use the same S' which exists due to (I), then there exists an A' depending on S' so that:

$$\mathsf{A}'^f\ \mathbf{br}\ G^f \wedge S'^{\mathsf{A}',f}\ \not\mathbf{br} f\ . \tag{3.2}$$

We notice that if now we add an oracle A' to S' that can be ignored by S' in (3.1), we also obtain that $[S'^{\mathsf{A}',f}\ \mathbf{br}\ f]$ which is a contradiction with (3.2). □

After we examine the result which was proved in [16], we start to use this approach to prove other two arrows in Figure 3.3 as follows.

Theorem 2 (Equivalence of BNNa and BBNa) *For all primitives $\mathcal{P}$ and $\mathcal{Q}$, there is a BBNa-reduction for efficient adversaries A if and only if there is a BNNa-reduction.*

Proof. Using straightforward logical deductions, it follows that BBNa-reductions imply BNNa-reductions. Now we want to prove converse direction. First we assume that there is a BNNa-reduction (the order of the quantifiers is $\exists G\forall f\forall A\exists S$) between the primitives $\mathcal{P}$ and $\mathcal{Q}$, but there is no BBNa-reduction. From the Table-3.2 we know that it means for any poly-time oracle machines G, there exists an $f\in\mathcal{Q}$ such that for all poly-time oracle machines S there exists an efficient adversary A(the order of the quantifiers is $\forall G\exists f\forall S\exists A$) such that $[\mathsf{A}^f\ \mathbf{br}\ G^f\wedge S^{A,f}\ \not\mathbf{br} f]$. From the assumption we can pick a fixed G then pick an f which depends on G such that:

$$\text{(I)}\ \underset{\text{pol}}{\forall}\,\mathsf{A}\,\underset{\text{pol}}{\exists}\,S\colon [\mathsf{A}^f\ \mathbf{br}\ G^f\Rightarrow S^f\ \mathbf{br}\ f]\ .$$

$$\text{(II)}\ \underset{\text{pol}}{\forall}\,S\,\underset{\text{pol}}{\exists}\,\mathsf{A}\colon [\mathsf{A}^f\ \mathbf{br}\ G^f\wedge S^{\mathsf{A},f}\ \not\mathbf{br} f]\ .$$

Like in the previous situation, For (I) let A satisfy $[\mathsf{A}^f\ \mathbf{br}\ G^f]$, then there exists an S' which depends on A such that:

$$\underset{\text{pol}}{\exists}\,S'\colon [S'^f\ \mathbf{br}\ f]\ . \tag{3.3}$$

For case (II) we use the same S' which exists due to (I), then there exists an A' depending on S' so that:

$$\mathsf{A}'^{f}\,\mathbf{br}\,G^{f} \wedge S'^{A',f} \not\mathbf{br} f \quad . \tag{3.4}$$

We notice that if now we add an oracle A' to S' that can be ignored by S' in (3.3), we also obtain that $[S'^{\mathsf{A}',f}\,\mathbf{br}\,f]$ which is a contradiction with (3.4). □

From the proof, we can understand that the condition for Arrow 1 and Arrow 2 are quite similar, but when we try to use the same approach to prove Arrow 3 we are not successful.

Using straightforward logical deductions, it also follows that BBBa-reductions imply BNBa-reductions. Now we want to prove converse direction. We use the same approach first assume that there is a BNBa-reduction(the order of the quantifiers is $\exists G \forall A \exists S \forall f$) between the primitives $\mathcal{P}$ and $\mathcal{Q}$, but there is no BBBa-reduction. From the Table-3.2 we know that it means that for any poly-time oracle machines G and poly-time oracle machines S, there exists an $f \in \mathcal{Q}$, and an efficient adversary A(the order of the quantifiers is $\forall G \forall S \exists A \exists f$) such that $[\mathsf{A}^{f}\,\mathbf{br}\,G^{f} \wedge S^{A,f} \not\mathbf{br} f]$. From the assumption we pick a fixed G such that:

$$\text{(I)} \underset{\text{pol}}{\forall}\mathsf{A}\underset{\text{pol}}{\exists}S\,\forall f\colon [\mathsf{A}^{f}\,\mathbf{br}\,G^{f} \Rightarrow S^{f}\,\mathbf{br}\,f] \quad .$$

$$\text{(II)} \underset{\text{pol}}{\forall}S\underset{\text{pol}}{\exists}\mathsf{A}\,\exists f\colon [\mathsf{A}^{f}\,\mathbf{br}\,G^{f} \wedge S^{\mathsf{A},f} \not\mathbf{br} f] \quad .$$

Here we use the same idea in Theorem 1. For (I) let an adversary A and f satisfy $[\mathsf{A}^{f}\,\mathbf{br}\,G^{f}]$ then there exists a S' which depends on A such that $\underset{\text{pol}}{\exists}S'\colon (S'^{f}\,\mathbf{br}\,f)$(*).
For case (II) we use the same S' which exists due to (I), then there exists an A' and f' such that both of them depend on S' and satisfy $[\mathsf{A}'^{f'}\,\mathbf{br}\,G^{f'} \wedge S'^{\mathsf{A}',f'} \not\mathbf{br} f']$(**). We notice that if now we add an adversary oracle A' to S' that can be ignored by S' in (*), we obtain that $[S'^{\mathsf{A}',f}\,\mathbf{br}\,f]$(***). By comparing(**) and (***) we consider the f and f' from a distribution $\mathcal{F}$ that S can break f with the access to f, but for f' not true. There is no contradiction result, hence it is probably not equivalent between Arrow 3.

To be sure the conjecture can only work for efficient adversaries in CAP framework. It is necessary to prove that for inefficient adversaries, BNN-reductions do not imply-BBN reductions, BNB-reductions do not imply BBB-reductions and NNN-reductions do not imply-NBN-reductions. The proof has been already done by Paul Baecher et al. [16]. Here we just give a brief review on the related proof as follows.

Theorem 3 *For all primitives $\mathcal{P}$ and $\mathcal{Q}$, if there is a BBB-reduction, there is a BNB-reduction. but the reverse is not true. Similarly, there is a BNN reduction, but no BBN reduction, as well as a NNN reduction, but no NBN reduction. [16]*

Proof. Existence of a reduction of one strong type can trivially imply all weaker types, hence if there is an BNB reduction, then there is also a BNN and a NNN reduction. To prove the Theorem 3, we just need to prove there is no NBN reduction, then we can get a simple deduction that neither a BBN, nor a BBB reduction exists. The most important fact is that for previous notions the reduction has to depend on the adversary in a non-black-box way, for later notions the reduction has to be universal for all adversaries in a black-box way. Paul Baecher et al. [16] show that Goldreich–Levin reduction(a BNB reduction) has to depend on the adversary's success probability. Furthermore, they show that there is no NBN reduction from $\mathcal{P}$ to $\mathcal{Q}$. As the length of the paper is limited, we omit the detail of the proof.

□

From all the things above, we notice the conjecture about there is no inherent restriction in treating the adversary as a black-box may only work for some notions like XNNa and XBNa, but probably not work for XBBa and XNBa, where $X \in \{\text{B,N}\}$. In other words, this means that in the reduction condition, the quantifier $\exists S$ has to stand after the quantifier $\forall f$. Based on all the result above, we can get a new view for the CAP framework as Figure 3.4

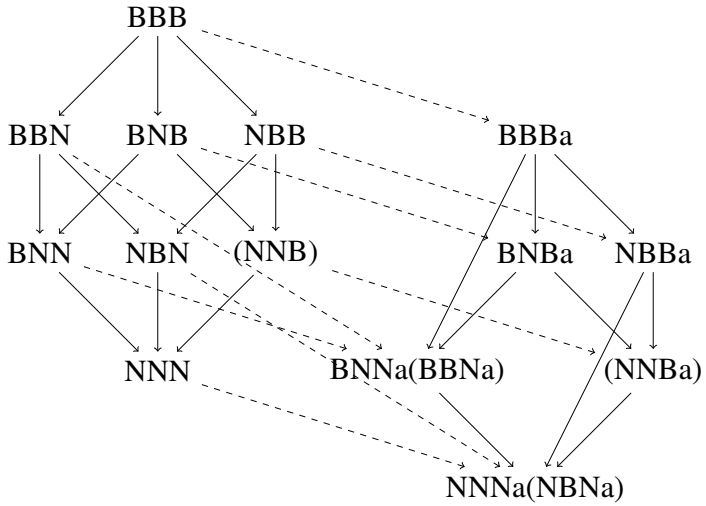

Figure 3.4: The new view for the CAP framework. An arrow goes from a more restricted form of reduction to a less restricted one. The dashed arrows designate implications

3.4 Parametrized Black-Box Reductions

If we consider a black-box reduction, we usually request the reduction algorithm cannot use any internals information of the adversary A except the input and output behavior. Strictly speaking, this information may include running time, number of queries, or the actual success probability of a given adversary. In practice the reduction, more or less, may depend on these information, we call this type of reduction as *parametrized black-box reductions*. Not surprisingly, most reductions can be considered as this type in real life. Paul Baecher et al. [16] also give two main notions of parametrized black-box reductions, namely, *parameter aware* reductions and *parameter dependent* reductions. Before we go into the details of each notion of parametrized black-box reductions, we need first consider two typical examples.

First let us consider an example of the reduction from collision-resistant functions to one-way functions described in [1]. Recall that the reduction algorithm S can choose random input x to f and receives some $f(x)$, then the reduction uses amplification techniques by accessing to the adversary to call $A(f(x))$ many times in order to get a second pre-image for $f(x)$. The adversary A succeeds with probability $\delta(k)$, the chance of not having a collision after using this method m times is about $(1 - \delta(k))^m$. As the amplification step heavily depends on probability of this success probability $\delta(k)$, the reduction is not universal for the adversary A anymore. Other than that, the reduction treats both the adversary and the primitive as black boxes. Note that in this case the reduction needs (one of) the parameters as explicit input, and we call this (black-box) reductions parameter aware.

Second, let us consider an example of a reduction from the unforgeable of a MAC scheme reduces to its own unforgeable described in [16]. We define a reduction algorithm S depends on queries and answers between the unforgeability game and the adversary A, whereas the code of the reduction algorithm itself is universal for all A. Specifically, the running time of the reduction depends on the security parameter and the number of queries placed by the adversary, although the reduction algorithm itself do not use any information except the input and output behavior of the adversary. Thus this type of reductions usually allow only an a priori limited number of interactions with the adversary. From this case, we know that we sometimes want to allow the reduction, especially its running time, to depend on adversarial parameters such as its number of queries, and we call this (black-box) reductions parameter dependent.

Here let us roughly compare the difference between the two notions of parametrized black-box reductions, in the parameter-dependent case we know that the running time of the reduction depends on adversarial parameters such as the number of queries by the adversary. In order to have fixed bounds on the running time of the reduction, we usually have restrictions for adversarial parameters. In the parameter aware case, the reduction receives some auxiliary information about the adversary such as the success probability of the adversary which may not be even known by the adversary itself, this is very similar to case of non-uniform model advice.

3.4.1 Relations

Obviously, parametrized black-box reductions and separations rely critically on the specific parameters of the adversary. And in this case we do not simply consider the adversary in a black-box way or non-black-box way, in other words, some of our result about the black-box use of efficient adversaries is equivalent to non-black-box use does not carry over to the parametrized case anymore. In contrast, the fundamental relation such as existence of a reduction of one strong type can trivially imply all weaker types still apply.

3.5 Poly-Preserving Reductions

A crucial property of a reduction from $\mathcal{P}$ to $\mathcal{Q}$ is how much security is maintained by the reduction. To measure this fairly, Buldas and Niitsoo [2] strengthened the guarantee condition and the class of reductions restricted in the following reasonable way:

Definition 21 (Poly-preserving reductions [3]) *A reduction of $\mathcal{P}$ to $\mathcal{Q}$ is* poly-preserving *if the security guarantee (S) decreases the advantage by at most a polynomial amount, i.e. there exists $c \geq 1$ (independent of f, A and k) such that:*

$$\mathrm{Adv}_k^{\mathcal{Q}}(S^f, f) \geq \left[\mathrm{Adv}_k^{\mathcal{P}}(A^f, G^f))\right]^c \quad . \tag{3.5}$$

Note that black-box reductions that satisfy Def. 21 for any certain (unspecified) c were first defined by Luby [27] and the condition (3.5) is a stronger form of the general

reduction condition:

$$\mathrm{ADV}_k^{\mathcal{P}}(A^f, G^f) \neq k^{-\omega(1)} \quad \Rightarrow \quad \mathrm{ADV}_k^{\mathcal{Q}}(S^f, f) \neq k^{-\omega(1)} \ . \tag{3.6}$$

Theorem 4 *The poly-preserving reduction condition (3.5) implies the general reduction condition (3.6), but the reverse is not true.*

Proof. if A^f breaks G^f, it means that $\mathrm{ADV}_k^{\mathcal{P}}(A^f, G^f) \neq k^{-\omega(1)}$, then we derive that $\mathrm{ADV}_k^{\mathcal{Q}}(S^f, f) \geq \left[\mathrm{ADV}_k^{\mathcal{P}}(A^f, G^f))\right]^c \neq (k^{-\omega(1)})^c \neq k^{-\omega(1)}$. This proves that the condition (3.5) implies the condition (3.6). Then we prove the reverse is not true, first we assume that $\mathrm{ADV}_k^{\mathcal{P}}(A^f, G^f) = k^{-1} \neq k^{-\omega(1)}$, and $\mathrm{ADV}_k^{\mathcal{Q}}(S^f, f) = k^{-c_f} \neq k^{-\omega(1)}$, here c depends on f and increases progressively. If $\forall c > 0\ \exists k\colon c_k \gg c$, then we derive that $\left[\mathrm{ADV}_k^{\mathcal{P}}(A^f, G^f))\right]^c = [k^{-1}]^c = k^{-c} > k^{-c_k} = \mathrm{ADV}_k^{\mathcal{Q}}(S^f, f)$ which is a contradiction with (3.5). □

4 Oracle Separation Methods

As we mentioned earlier, constructing complex primitives from simpler ones is one of the most fundamental questions in cryptography. However, some constructions, such as the construction of public-key encryption from one-way functions, is still elusive. Impagliazzo and Rudich [8] showed that constructions of key agreement (KA) based on one-way functions (OWP) imply a proof that $\mathsf{P} \neq \mathsf{NP}$. They were the first to prove arguments against the existence of black-box reductions. As the opposite of the reductions, separations can save a lot of wasted effort by guiding researchers away from hopeless approaches, they are also quite meaningful for cryptographers.

In this chapter we review the three main techniques for black-box separations. In Section 4.1, we briefly introduce the concept of meta-reductions, then we give a summary of the relations between different meta-reductions. In Section 4.2, we review the oracle-extraction based separation and give the separation conditions according to different types of black-box reductions. In Section 4.3, we introduce a new oracle separation approach, namely, averaging approach, then we give the separation conditions and the proofs for each separation condition.

4.1 Meta-Reductions

A meta-reduction is a "reduction which can be use to prove the separation result" which means if there exists a meta-reduction from a $(\mathcal{P} \rightarrow \mathcal{Q})$-reduction to a primitive N (which usually can be $\mathcal{Q}$ itself), then there is no reduction from $\mathcal{P}$ to $\mathcal{Q}$, if N exists. It turns out that meta-reductions may depend on the above notions for black-box reductions, so in this section, we first review the definition of meta-reductions based on the CAP framework in [16], then we try to briefly analyze the relations between them. We start from the BBB meta-reduction, the definition is as follows.

Definition 22 (BBB-Meta-Reductions) *We say that there is a* $[(\mathcal{P} \rightarrow \mathcal{Q}\text{-}BBB) \rightarrow N]$

meta reduction, if whenever there is a BBB-reduction from $\mathcal{P}$ to $\mathcal{Q}$, then for every $g \in N$, there is a poly-time M such that M^g breaks g.

The quantification of the security statement for the definition can be described as follows.

$$\underset{\text{pol}}{\exists}\, G \;\underset{\text{pol}}{\exists}\, S\, \forall f\, \forall \mathsf{A} : [(\mathsf{A}^f \,\mathbf{br}\, G^f) \Rightarrow (S^{\mathsf{A},f} \,\mathbf{br}\, f)] \Rightarrow \forall g \;\underset{\text{pol}}{\exists}\, M \colon (M^g \,\mathbf{br}\, g) \;\; .$$

Constructing a BBB-meta-reduction usually consist of following steps, First using an instance g of N to instantiates a f of $\mathcal{Q}$ (note that if N is $\mathcal{Q}$ itself, we do not need this step). Second design an all-powerful A which can breaks $\mathcal{P}$ and $\mathcal{Q}$, formally, we need find an (inefficient) adversary A that can break G^f as $\mathcal{P}$. then the efficient reduction algorithm S turn this A into a successful adversary to break the instance $f = g$. Third, replace the (inefficient) adversary A by the efficient meta-reduction M. This is usually done by carefully rewinding the reduction. As an example for BBB-meta-reductions, we consider a separation result so-called one-more problems proposed by Bresson et al. [24]. It describes that an adversary may query an oracle for solutions on n instances, but in order to be successful, the adversary has to provide eventually $n + 1$ instance. Bresson et al. find a meta-reduction through rewinding technique to show that solving such problems on n instances cannot reduce to the case of solving the same problem on $n - 1$ instances. As the reduction described in [24] treats the adversary and the primitive in a black-box way, and the construction treat the primitive in black box way as well. There is a BBB-meta-reduction according to our definition. We now turn to the definition of BNB-meta-reductions.

Definition 23 (BNB-Meta-Reductions) *We say that there is a $[(\mathcal{P} \rightarrow \mathcal{Q}$-BNB$)\rightarrow N]$ meta reduction, if whenever there is a BNB-reduction from $\mathcal{P}$ to $\mathcal{Q}$, then for every $g \in N$, there is a poly-time M such that M^g breaks g.*

The quantification of the security statement for the definition can be described as follows.

$$\underset{\text{pol}}{\exists}\, G\, \forall \mathsf{A}\; \underset{\text{pol}}{\exists}\, S\, \forall f : [(\mathsf{A}^f \,\mathbf{br}\, G^f) \Rightarrow (S^{\mathsf{A},f} \,\mathbf{br}\, f)] \Rightarrow \forall g \;\underset{\text{pol}}{\exists}\, M \colon (M^g \,\mathbf{br}\, g) \;\; .$$

If we compare the definition of BBB-meta-reductions and BNB-meta-reductions, we note that building a BNB-meta-reduction might be more difficult than building a BBB-meta-reduction. The reason is that in BNB-meta-reduction, the adversary A has to be universal

for all algorithm S, in other words, if we consider the meta-reduction as a separation result from $\mathcal{P}$ to $\mathcal{Q}$, the BNB-meta-reduction can trivially imply BBB-meta-reduction. We now continue to give the definition of BBN-meta-reductions

Definition 24 (BBN-Meta-Reductions) *We say that there is a* $[(\mathcal{P} \rightarrow \mathcal{Q}\text{-}BBN) \rightarrow N]$ *meta reduction, if whenever there is a BBN-reduction from* $\mathcal{P}$ *to* $\mathcal{Q}$*, then for every* $g \in N$*, there is a poly-time* M *such that* M^g *breaks* g*.*

The quantification of the security statement for the definition can be described as follows.

$$\underset{\text{pol}}{\exists} G \, \forall f \; \underset{\text{pol}}{\exists} S \, \forall \mathsf{A} : [(\mathsf{A}^f \, \mathbf{br} \, G^f) \Rightarrow (S^{\mathsf{A},f} \, \mathbf{br} \, f)] \Rightarrow \forall g \; \underset{\text{pol}}{\exists} M : (M^g \, \mathbf{br} \, g) \ .$$

Definition 25 (NBB-Meta-Reductions) *We say that there is a* $[(\mathcal{P} \rightarrow \mathcal{Q}\text{-}NBB) \rightarrow N]$ *meta reduction, if whenever there is a NBB-reduction from* $\mathcal{P}$ *to* $\mathcal{Q}$*, then for every* $g \in N$*, there is a poly-time* M *such that* M^g *breaks* g*.*

The quantification of the security statement for the definition can be described as follows.

$$\underset{\text{pol}}{\exists} S \, \forall f \; \underset{\text{pol}}{\exists} G \, \forall \mathsf{A} : [(\mathsf{A}^f \, \mathbf{br} \, G^f) \Rightarrow (S^{\mathsf{A},f} \, \mathbf{br} \, f)] \Rightarrow \forall g \; \underset{\text{pol}}{\exists} M : (M^g \, \mathbf{br} \, g) \ .$$

Definition 26 (BNN-Meta-Reductions) *We say that there is a* $[(\mathcal{P} \rightarrow \mathcal{Q}\text{-}BNN) \rightarrow N]$ *meta reduction, if whenever there is a BNN-reduction from* $\mathcal{P}$ *to* $\mathcal{Q}$*, then for every* $g \in N$*, there is a poly-time* M *such that* M^g *breaks* g*.*

The quantification of the security statement for the definition can be described as follows.

$$\underset{\text{pol}}{\exists} G \, \forall f \, \forall \mathsf{A} \; \underset{\text{pol}}{\exists} S : [(\mathsf{A}^f \, \mathbf{br} \, G^f) \Rightarrow (S^{\mathsf{A},f} \, \mathbf{br} \, f)] \Rightarrow \forall g \; \underset{\text{pol}}{\exists} M : (M^g \, \mathbf{br} \, g) \ .$$

Definition 27 (NBN-Meta-Reductions) *We say that there is a* $[(\mathcal{P} \rightarrow \mathcal{Q}\text{-}NBN) \rightarrow N]$ *meta reduction, if whenever there is a NBN-reduction from* $\mathcal{P}$ *to* $\mathcal{Q}$*, then for every* $g \in N$*, there is a poly-time* M *such that* M^g *breaks* g*.*

The quantification of the security statement for the definition can be described as follows.

$$\forall f \underset{\text{pol}}{\exists} G \underset{\text{pol}}{\exists} S \, \forall \mathsf{A} : [(\mathsf{A}^f \, \mathbf{br} \, G^f) \Rightarrow (S^{\mathsf{A},f} \, \mathbf{br} \, f)] \Rightarrow \forall g \underset{\text{pol}}{\exists} M : (M^g \, \mathbf{br} \, g) \ .$$

As an example for NBN-meta-reductions, we consider the separation result from blind signatures to hard non-interactive problem proposed by Fischlin and Schröder [23]. We define the primitive $\mathcal{P}$ as a (blind) signature scheme and the primitive $\mathcal{Q}$ as a hard non-interactive problem. Note that the primitive N is equal to the primitive $\mathcal{Q}$ here, so we do not need instantiates a f via g. We need find an adversary A which can compute a secret key sk from pk to break G^f as $\mathcal{P}$, then let it query the signature oracle to collect signatures, finally let adversary A compute a forgery to break f as $\mathcal{Q}$. After constructing an all-powerful adversary A that is successful against the primitive f. We rewinding the reduction at appropriate places in the query phase to find a meta-reduction M^f which can simulating the behavior $S^{A,f}$ efficiently. As a consequence, M^f can break the primitive f as $\mathcal{Q}$. Here, the construction is non black box, the adversary is treated as a black box, but the reduction is not restricted to black-box access to the primitive. So it classifies as a NBN-meta-reduction.

Another example of NBN-meta-reductions is the work by Pass [25], which shows that a certain type of argument system cannot reduction to certain standard assumptions. Specifically they restating several constructions including the Schnorr identification scheme, the adaptive selective decommitment problem and unique blind signatures as an argument system, then they use meta reduction to show that these constructions cannot be based on standard assumptions. Since these results hold when adversary was treated as a black box by the reduction, but both of the construction and the reduction may depend on the standard assumptions, this type of meta reduction is also considered as a NBN-meta-reduction according to our definition.

Definition 28 (NNN-Meta-Reductions) *We say that there is a* $[(\mathcal{P} \rightarrow \mathcal{Q}\text{-}NNN) \rightarrow N]$ *meta reduction, if whenever there is a NNN-reduction from* $\mathcal{P}$ *to* $\mathcal{Q}$*, then for every* $g \in N$*, there is a poly-time* M *such that* M^g *breaks* g*.*

The quantification of the security statement for the definition can be described as follows.

$$\forall f \underset{\text{pol}}{\exists} G \, \forall \mathsf{A} \underset{\text{pol}}{\exists} S : [(\mathsf{A}^f \, \mathbf{br} \, G^f) \Rightarrow (S^{\mathsf{A},f} \, \mathbf{br} \, f)] \Rightarrow \forall g \underset{\text{pol}}{\exists} M : (M^g \, \mathbf{br} \, g) \ .$$

Note that all introduced notions for meta reductions easily translate into meta reductions

for efficient adversaries by only quantifying over efficient A. So we avoid to give repetitive definitions here.

4.1.1 Relations between Meta-Reductions

Meta-reduction can be considered as separation result from primitive $\mathcal{P}$ to primitive $\mathcal{Q}$, the whole picture of the meta-reduction in CAP is still a partial order hierarchy, whereas NNN-meta-reductions become the strongest restricted and BBB–meta-reductions be the weakest. In particular it is pictured as an inverted image of the original CAP framework as show in Figure 4.1.

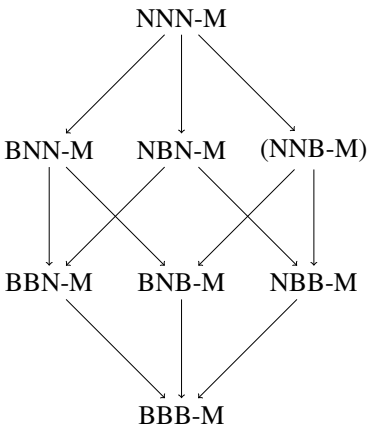

Figure 4.1: Meta-reductions in CAP framework and the relation between notions of reduction, M stand for meta-reduction

Theorem 5 *If R_1-type reduction implies R_2-type reduction, then $(R_2 \rightarrow M)$-mata reduction implies $(R_1 \rightarrow M)$-mata reduction, $R_1, R_2 \in \{$**BBB, BNB, BBN, NBB, BNN, NBN, NNN**$\}$*

Proof. if $R_1 \Rightarrow R_2$, then according to the logical deduction it follows that $R_2 \Rightarrow M$ implies $R_1 \Rightarrow M$. □

Theorem 6 *If primitive N exists and $(R_2 \rightarrow M)$-mata reduction implies $(R_1 \rightarrow M)$-mata reduction, then R_1-type reduction implies R_2-type reduction, $R_1, R_2 \in \{$**BBB, BNB, BBN, NBB, BNN, NBN, NNN**$\}$*

Proof. if $\neg M \wedge (R_2 \Rightarrow M) \Rightarrow (R_1 \Rightarrow M)$, then according to the logical deduction it follows that $\neg R_2 \Rightarrow \neg R_1$ which implies $R_1 \Rightarrow R_2$. □

Note that meta-reduction depend on the above CAP notions. They still somehow keep the same property in relations such as existence of a reduction of one strong type can trivially imply the existence of reductions of all weaker types. What we are particularly interested in our previous result for equivalence of black-box use and non-black-box use of efficient adversaries is still imply to meta-reduction or not. Here we also give a proof of equivalence of BNNa-meta-reduction and BBNa-meta-reduction as example.

Theorem 7 (Equivalence of BNNa-meta-reduction and BBNa-meta-reduction) *For all primitives $\mathcal{P}$ and $\mathcal{Q}$, there is a BNNa-meta-reduction for efficient adversaries* A *if and only if there is a BBNa-meta-reduction.*

Proof. From Theorem 2 we know that it is equivalence of BNNa-reduction and BBNa-reduction, then according to Theorem 5 it follows that BBNa-meta-reduction and BNNa-meta-reduction are also equivalent.

□

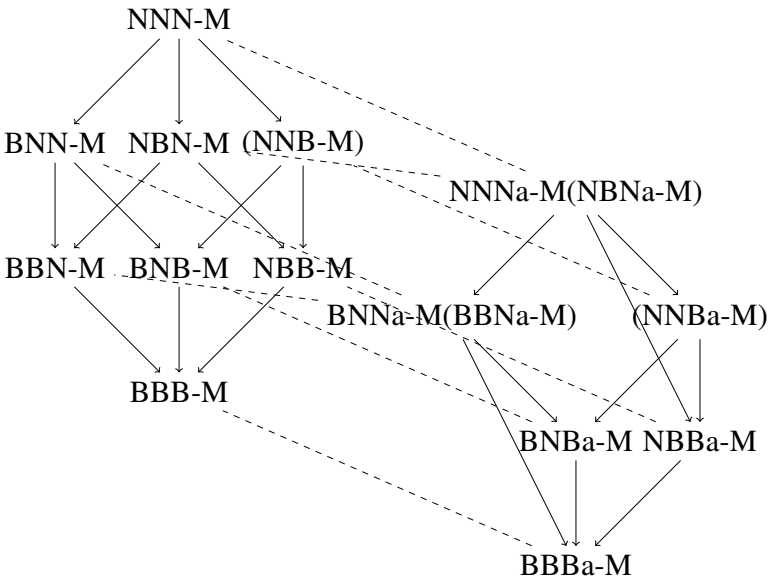

Figure 4.2: The relations between meta-reductions for (in)efficient adversaries. An arrow goes from a more restricted form of reduction to a less restricted one. The dashed arrows designate implications

The proof for NNNa-meta-reduction and NBNa-meta-reduction is quite similar to the proof of equivalence of BNNa-meta-reduction and BBNa-meta-reduction, so we omit the detail of the proof. From all the things above, we notice that meta-reductions are the separation of the reductions, they do not change our results of previous proof. The Figure 4.2 shows the image of the relations between meta-reductions.

4.2 Oracle-Extraction Based Separation

In this section of the the thesis, we use the CAP framework to extend the notions for oracle-extraction based separation which were not covered in [3]. Specifically, first we review the necessary fundamental about how to get a separation from the oracle-extraction, then we propose an extended table based on the work by Buldas and Niitsoo [3]. Finally, in Section 4.2.1 we give proofs for each new separation condition according to the extended table.

In cryptography, a reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$ usually means that either $\mathcal{P}$ exists or $\mathcal{Q}$ does not exist. As a negative result, to show there is no black-box reductions from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, we need prove that $\mathcal{P}$ does not exist, but at the same time, $\mathcal{Q}$ exists. Further, if we want to show that there is no black-box reductions from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$ by using oracle separation, we need to define an oracle $\mathcal{O}$ and show a *breakage argument* that there is no secure $\mathcal{P}$ relative to $\mathcal{O}$, but at the same time, there exists a *security argument* that a secure $\mathcal{Q}$ relative to $\mathcal{O}$. In classical separation results of complexity theory, oracles are defined as fixed functions and have specific behavior. In cryptographic separations, it is very difficult to define a specific separation oracle. For instance, it is hard to show that one-way functions exist relative to an oracle, because we even do not know whether one-way functions exist in the standard computational model. So instead of defining a fixed oracle we need define a certain probability distribution then assume that oracles are chosen randomly from that probability distribution. Hence the most important thing for an oracle separation is to define a probability distribution $\mathcal{F}$, then randomly choose a $\mathcal{O}$ from $\mathcal{F}$ to show that there is secure instance $f^{\mathcal{O}}$ of $\mathcal{Q}$, but no instance $G^{\mathcal{O}}$ of $\mathcal{P}$ is secure relative to $\mathcal{O}$. According to the oracle embedding techniques introduced by Simon [10], we know that the secure instance f of $\mathcal{Q}$ can be identified with the oracle $\mathcal{O}$, so we use the oracle distribution $f \leftarrow \mathcal{F}$ instead of $\mathcal{O} \leftarrow \mathcal{F}$. Here we use a NNNa-redcution as an example to show how to get a separation by using the oracle-extraction based separation techniques. First we need to prove the breakage argument in the oracle-extraction based separation techniques. We have to show that:

- b_1: For every instance G^f of $\mathcal{P}$, there is a poly-time machine A such that A^f breaks G^f with overwhelming probability,
  i.e. $\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A\colon \underset{f \leftarrow \mathcal{F}}{\mathbf{E}} \left[\textsc{Adv}_k(A^f, G^f) \right] = 1 - k^{-\omega(1)}$.

- b_2: According to b_1 and Lemma 7, we know that for measure one of f's and every instance G^f of $\mathcal{P}$, there is a poly-time A, so that A^f breaks G^f,

i.e. $\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A\colon \Pr_{f\leftarrow\mathcal{F}}\left[\text{ADV}_k(A^f, G^f) = 1 - k^{-\omega(1)}\right] = 1$.

b_3: According to b_2 and Lemma 5, we know that for measure one of oracles f and every G^f, there is a poly-time machine A such that A^f breaks G^f,
i.e. $\Pr_{f\leftarrow\mathcal{F}}\left[\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A\colon \text{ADV}_k(A^f, G^f) = 1 - k^{-\omega(1)}\right] = 1$.

Note that according to Lemma 8, we cannot use the weaker statement $\mathbf{E}_{f\leftarrow\mathcal{F}}\left[\text{ADV}_k(A^f, G^f)\right] = k^{-O(1)}$ to replace the statement (b_1), because there exists a counterexample for which $\text{ADV}_k(A^f, G^f) = k^{-\omega(1)}$ for all f. Hence this weaker statement cannot imply that there is an f for which $\text{ADV}_k(A^f, G^f) = k^{-O(1)}$.

Second we need to prove the security argument in the oracle-extraction based separation techniques. We have to show that:

s_1: An instance f of $\mathcal{Q}$ can be broken by every fixed poly-time adversary S that uses f as an oracle only with negligible success, on average,
i.e. $\underset{\text{pol}}{\forall} S\colon \mathbf{E}_{f\leftarrow\mathcal{F}}\left[\text{ADV}_k(S^f, f)\right] = k^{-\omega(1)}$.

s_2: According to s_1 and Lemma 6, for measure one of f's, no poly time S can break f,
i.e. $\underset{\text{pol}}{\forall} S\colon \Pr_{f\leftarrow\mathcal{F}}\left[\text{ADV}_k(S^f, f) = k^{-\omega(1)}\right] = 1$.

s_3: According to s_2 and Lemma 5, for measure one of oracles f, no poly-time S can break f better than with negligible success,
i.e. $\Pr_{f\leftarrow\mathcal{F}}\left[\underset{\text{pol}}{\forall} S\colon \text{ADV}_k(S^f, f) = k^{-\omega(1)}\right] = 1$.

Finally, from all the statements above we can get that measure one of oracles satisfy both the breakage and the security argument, then according to the Lemma 4 (Probabilistic Argument) there exists a fixed separation oracle for which the statements also hold. As there is no need to have countability argument for our statements s_1, s_2, b_1 and b_2, they can still apply to non-uniform reductions, but the steps s_3 and b_3 do not, because there are uncountably many non-uniform machines S and G.

In Table 4.1, we extend the original table proposed by Buldas and Niitsoo [3] via listing the separation conditions for other three types of reductions in the CAP framework. All the proofs of new types are given in the next section. The important result we obtain from Table 4.1 is that the non-existence of BBB and the BNBa reductions can be proven without the countability argument. The main reason is that in the separation condition the oracle distribution $\mathcal{F}$ may depend on G and S. Formally, this means that in the separation

Table 4.1: Reduction types and separation conditions for oracle extraction based separations. The notions with underline already been given by Buldas and Niitsoo [3]

Type	Reduction Condition	Separation Condition
BBB-Reduction	$\underset{\text{pol}}{\exists}\, G\, \underset{\text{pol}}{\exists}\, S \forall f \forall A\colon$ $A \,\mathbf{br}\, G^f \Rightarrow S^{A,f} \,\mathbf{br}\, f$	$\underset{\text{pol}}{\forall}\, G\, \underset{\text{pol}}{\forall}\, S \exists \mathcal{F}\colon \underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^{f,A}, f)\right] = k^{-\omega(1)}$
BNBa-Reduction (Strong Semi bb)	$\underset{\text{pol}}{\exists}\, G\, \underset{\text{pol}}{\forall}\, A\, \underset{\text{pol}}{\exists}\, S \forall f\colon$ $A^f \,\mathbf{br}\, G^f \Rightarrow S^f \,\mathbf{br}\, f$	$\underset{\text{pol}}{\forall}\, G\, \underset{\text{pol}}{\exists}\, A\, \underset{\text{pol}}{\forall}\, S \exists \mathcal{F}\colon \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^f, f)\right] = k^{-\omega(1)}$
BBNa-Reduction	$\underset{\text{pol}}{\exists}\, G \forall f\, \underset{\text{pol}}{\exists}\, S\, \underset{\text{pol}}{\forall}\, A\colon$ $A^f \,\mathbf{br}\, G^f \Rightarrow S^{A,f} \,\mathbf{br}\, f$	$\underset{\text{pol}}{\forall}\, G \exists \mathcal{F}\, \underset{\text{pol}}{\forall}\, S\, \underset{\text{pol}}{\exists}\, A\colon \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^{A,f}, f)\right] = k^{-\omega(1)}$ Countability argument for S
BNNa-Reduction (Weak Semi bb)	$\underset{\text{pol}}{\exists}\, G\, \underset{\text{pol}}{\forall}\, A \forall f\, \underset{\text{pol}}{\exists}\, S\colon$ $A^f \,\mathbf{br}\, G^f \Rightarrow S^f \,\mathbf{br}\, f$	$\underset{\text{pol}}{\forall}\, G\, \underset{\text{pol}}{\exists}\, A \exists \mathcal{F}\colon \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{\text{pol}}{\forall}\, S\, \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^f, f)\right] = k^{-\omega(1)}$ Countability argument for S
NBBa-Reduction	$\underset{\text{pol}}{\exists}\, S \forall f\, \underset{\text{pol}}{\exists}\, G\, \underset{\text{pol}}{\forall}\, A\colon$ $A^f \,\mathbf{br}\, G^f \Rightarrow S^{A,f} \,\mathbf{br}\, f$	$\underset{\text{pol}}{\forall}\, S \exists \mathcal{F}\, \underset{\text{pol}}{\forall}\, G\, \underset{\text{pol}}{\exists}\, A\colon \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^{A,f}, f)\right] = k^{-\omega(1)}$ Countability arguments for G
NBNa-Reduction	$\forall f\, \underset{\text{pol}}{\exists}\, G\, \underset{\text{pol}}{\exists}\, S\, \underset{\text{pol}}{\forall}\, A\colon$ $A^f \,\mathbf{br}\, G^f \Rightarrow S^{A,f} \,\mathbf{br}\, f$	$\exists \mathcal{F}\, \underset{\text{pol}}{\forall}\, G\, \underset{\text{pol}}{\forall}\, S\, \underset{\text{pol}}{\exists}\, A\colon \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^{A,f}, f)\right] = k^{-\omega(1)}$ Countability arguments for G and S
NNNa-Reduction (Variable Semi bb)	$\forall f\, \underset{\text{pol}}{\exists}\, G\, \underset{\text{pol}}{\forall}\, A\, \underset{\text{pol}}{\exists}\, S\colon$ $A^f \,\mathbf{br}\, G^f \Rightarrow S^f \,\mathbf{br}\, f$	$\exists \mathcal{F}\colon \underset{\text{pol}}{\forall}\, G\, \underset{\text{pol}}{\exists}\, A\, \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ $\underset{\text{pol}}{\forall}\, S\, \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\textsc{Adv}_k(S^f, f)\right] = k^{-\omega(1)}$ Countability arguments for G and S

condition, the quantifier $\exists\mathcal{F}$ need stand after the quantifiers $\underset{\text{pol}}{\forall} G$ and $\underset{\text{pol}}{\forall} S$.

In sum, the oracle extraction based separation techniques are applicable to the *BNBa-reductions* and the *BBB-reductions* but not for other reductions, because most of them need the countability argument.

4.2.1 Proofs for Additional Oracle Extraction-Based Separations

We begin with the proof for nonexistence of a BBNa-reductions, the idea of the proof originates from Buldas and Niitsoo [3]. Note that in this proof we need the countability argument for S, because the reduction S may depend on f, More formally, in the separation condition, the quantifier $\exists\mathcal{F}$ now stands before the quantifiers $\underset{\text{pol}}{\forall} S$.

Theorem 8 *If* $\underset{pol}{\forall} G \exists\mathcal{F} \underset{pol}{\forall} S \underset{pol}{\exists} A\colon$ *so that (I)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ *and (II)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(S^{f,A}, f)\right] = k^{-\omega(1)}$*, there exist no uniform BBNa-Reductions.*

Proof. The overwhelming average argument for (I) and the negligible average argument for (II) imply:

$$\underset{\text{pol}}{\forall} S \underset{\text{pol}}{\exists} A\colon \Pr_{f\leftarrow\mathcal{F}}\left[A^f \mathbf{br} G^f\right] = 1 \wedge \Pr_{f\leftarrow\mathcal{F}}\left[S^{f,A} \not\mathbf{br} f\right] = 1 \ .$$

Now by the countability argument for S and Lemma 5 we obtain that:

$$\underset{\text{pol}}{\forall} G \exists\mathcal{F}\colon \Pr_{f\leftarrow\mathcal{F}}\left[\underset{\text{pol}}{\exists} S \underset{\text{pol}}{\forall} A\colon\ A^f \mathbf{br} G^f \wedge S^{f,A} \not\mathbf{br} f\right] = 1 \ .$$

By the Lemma 4 we have the negation of the BBNa-reduction:

$$\underset{\text{pol}}{\forall} G \exists f \underset{\text{pol}}{\forall} S \underset{\text{pol}}{\exists} A\colon\ \left[A^f \mathbf{br} G^f \wedge S^{f,A} \not\mathbf{br} f\right] \ .$$

□

Next, we give the proof for nonexistence of NBBa-reductions. As in the separation condition, the quantifier $\exists\mathcal{F}$ now stand before the quantifiers $\underset{\text{pol}}{\forall} G$. we need the countability argument for G.

Theorem 9 *If* $\underset{pol}{\forall} S \exists\mathcal{F} \underset{pol}{\forall} G \underset{pol}{\exists} A\colon$ *so that (I)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ *and (II)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(S^{f,A}, f)\right] = k^{-\omega(1)}$*, there exist no uniform NBBa-Reductions.*

Proof. The overwhelming average argument for (I) and the negligible average argument for (II) imply:

$$\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A\colon \ \Pr_{f\leftarrow\mathcal{F}}\left[A^f \,\mathbf{br}\, G^f\right] = 1 \wedge \Pr_{f\leftarrow\mathcal{F}}\left[S^{f,A} \not\mathbf{br} f\right] = 1 \ .$$

Now by the countability argument for G and Lemma 5 , we can obtain the similar result as:

$$\underset{\text{pol}}{\forall} S \exists \mathcal{F}\colon \Pr_{f\leftarrow\mathcal{F}}\left[\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A\colon\ A^f \,\mathbf{br}\, G^f \wedge\colon\ S^{f,A} \not\mathbf{br} f\right] = 1 \ .$$

By the Lemma 4 we have the negation of the NBBa-reduction:

$$\underset{\text{pol}}{\forall} S \exists f \underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A\colon \ \left[A^f \,\mathbf{br}\, G^f \wedge S^{f,A} \not\mathbf{br} f\right] \ .$$

□

Finally, we give the proof for nonexistence of NBNa-Reductions, As in the separation condition, the quantifier $\exists\mathcal{F}$ now stand before the quantifiers $\underset{\text{pol}}{\forall} G$ and $\underset{\text{pol}}{\forall} S$. we need the countability argument for both G and S.

Theorem 10 *If* $\exists\mathcal{F} \underset{pol}{\forall} G \underset{pol}{\forall} S \underset{pol}{\exists} \mathsf{A}\colon$ *so that (I)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(A^f, G^f)\right] = 1 - k^{-\omega(1)}$ *and (II)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(S^{f,\mathsf{A}}, f)\right] = k^{-\omega(1)}$*, there exist no uniform NBNa-Reductions.*

Proof. The overwhelming average argument for (I) and the negligible average argument for (II) imply that:

$$\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\forall} S \underset{\text{pol}}{\exists} A\colon \ \Pr_{f\leftarrow\mathcal{F}}\left[A^f \,\mathbf{br}\, G^f\right] = 1 \wedge \Pr_{f\leftarrow\mathcal{F}}\left[S^{f,A} \not\mathbf{br} f\right] = 1 \ .$$

Now by Lemma 5 and the countability argument for G and S, we obtain that:

$$\exists\mathcal{F}\colon \Pr_{f\leftarrow\mathcal{F}}\left[\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\forall} S \underset{\text{pol}}{\exists} \mathsf{A}\colon\ A^f \,\mathbf{br}\, G^f \wedge S^{f,A} \not\mathbf{br} f\right] = 1 \ .$$

By the Lemma 4 we have the negation of the NBNa-reduction:

$$\exists f \underset{\text{pol}}{\forall} G \underset{\text{pol}}{\forall} S \underset{\text{pol}}{\exists} A\colon \ \left[A^f \,\mathbf{br}\, G^f \wedge S^{f,A} \not\mathbf{br} f\right] \ .$$

□

4.3 Averaging-Based Separation

In this section of the the thesis, we study the oracle separation approach that is called *averaging approach* first introduced by Buldas and Niitsoo [2]. We use the CAP framework to extend the notions for averaging-based separation which were not covered in their paper. Similar to the last chapter, first we review the necessary background about how to get a separation through the averaging approach, then we propose an extended table for averaging-based separation based on the work [3]. Finally, we give proofs for each new separation condition according to the extended table in Section 4.3.2.

As we know many practical primitives are required to be secure in the non-uniform security model. It is clear that oracle extraction cannot be used in the non-uniform security model, because most of the reductions need the countability argument. In order to solve this kind of problem, Buldas and Niitsoo [2] introduce the averaging approach where the oracle extraction step is not necessary. They proved that the averaging approach is capable of showing that there are no non-uniform reductions between two primitives. Here we start to describe the idea of the averaging approach.

From the Table-4.1 we know that the reduction condition is deterministic and has the specific form as follows:

$$\textsc{Adv}_k^{\mathcal{P}}(A, G^f) \neq k^{-\omega(1)} \quad \Rightarrow \quad \textsc{Adv}_k^{\mathcal{Q}}(S^{f,A}, f) \neq k^{-\omega(1)} \ . \tag{4.1}$$

Both the security assumption and the breakage assumption are probabilistic in practical separations. i.e. involve an average success over the oracle. To show that there are no BBB-reductions from primitive $\mathcal{P}$ to primitive $\mathcal{Q}$, we have to derive a contradiction based on the reduction condition (4.1) and the separation conditions:

$$\begin{aligned} &\text{(S)} \ \mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}} \left[\textsc{Adv}_k^{\mathcal{P}}(A, G^f)\right] = 1 - k^{-\omega(1)} \\ &\text{(B)} \ \mathop{\forall}_{\text{pol}} S\colon \ \mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}} \left[\textsc{Adv}_k^{\mathcal{Q}}(S^{f,A}, f)\right] = k^{-\omega(1)} \ . \end{aligned}$$

If we consider the traditional approach such as oracle extraction, we need first focus on conditions (S) and (B) and try to extract fixed oracles f and A from $\mathcal{F}$ so that (4.1) is not satisfied. The average-based separation technique [2] uses another way. Formally, it first

Table 4.2: Reduction types and separation conditions for averaging-based separation in the case of poly-preserving black-box reductions. The notions with underline already been given by Buldas and Niitsoo [3]

Type	Reduction Condition	Separation Condition
<u>BBB</u>	$\underset{\text{pol}}{\exists} G \underset{\text{pol}}{\exists} S \forall f \forall A$: $\mathrm{ADV}_k(S^{f,A}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f)\right]^c$	$\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\forall} S \exists \mathcal{F}$: $\underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A, G^f)\right] \neq k^{-\omega(1)}$ $\underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^{f,A}, f)\right] = k^{-\omega(1)}$
<u>BNBa</u>	$\underset{\text{pol}}{\exists} G \underset{\text{pol}}{\forall} A \underset{\text{pol}}{\exists} S \forall f$: $\mathrm{ADV}_k(S^f, f) \geq \left[\mathrm{ADV}_k(A^f, G^f)\right]^c$	$\underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A \underset{\text{pol}}{\forall} S \exists \mathcal{F}$: $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A^f, G^f)\right] \neq k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^f, f)\right] = k^{-\omega(1)}$
BBNa	$\underset{\text{pol}}{\exists} G \forall f \underset{\text{pol}}{\exists} S_f \underset{\text{pol}}{\forall} A$: $\mathrm{ADV}_k(S_f^{f,A}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f)\right]^c$	$\underset{\text{pol}}{\forall} G \exists \mathcal{F} \underset{\text{pol}}{\forall} S \underset{\text{of}}{\forall} \varphi \underset{\text{pol}}{\exists} A$: $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A^f, G^f)\right] \neq k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^{f,A}_{\varphi(f)}, f)\right] = k^{-\omega(1)}$
<u>BNNa</u>	$\underset{\text{pol}}{\exists} G \forall f \underset{\text{pol}}{\forall} A \underset{\text{pol}}{\exists} S_f$: $\mathrm{ADV}_k(S_f^f, f) \geq \left[\mathrm{ADV}_k(A^f, G^f)\right]^c$	$\underset{\text{pol}}{\forall} G \exists \mathcal{F}$: $\underset{\text{pol}}{\exists} A \underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A^f, G^f)\right] \neq k^{-\omega(1)}$ $\underset{\text{pol}}{\forall} S \underset{\text{of}}{\forall} \varphi \underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^f_{\varphi(f)}, f)\right] = k^{-\omega(1)}$
NBBa	$\underset{\text{pol}}{\exists} S \forall f \underset{\text{pol}}{\exists} G_f \underset{\text{pol}}{\forall} A$: $\mathrm{ADV}_k(S^{f,A}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f_f)\right]^c$	$\underset{\text{pol}}{\forall} S \underset{\text{of}}{\forall} \psi \exists \mathcal{F} \underset{\text{pol}}{\forall} G \underset{\text{pol}}{\exists} A$: $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A^f, G^f_{\psi(f)})\right] \neq k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^{f,A}, f)\right] = k^{-\omega(1)}$
NBNa	$\forall f \underset{\text{pol}}{\exists} G_f \underset{\text{pol}}{\exists} S_f \underset{\text{pol}}{\forall} A$: $\mathrm{ADV}_k(S_f^{f,A}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f_f)\right]^c$	$\underset{\text{of}}{\forall} \psi \exists \mathcal{F} \underset{\text{pol}}{\forall} G \underset{\text{pol}}{\forall} S \underset{\text{of}}{\forall} \varphi \underset{\text{pol}}{\exists} A$: $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A^f, G^f_{\psi(f)})\right] \neq k^{-\omega(1)}$ $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^{f,A}_{\varphi(f)}, f)\right] = k^{-\omega(1)}$
<u>NNNa</u>	$\forall f \underset{\text{pol}}{\exists} G_f \underset{\text{pol}}{\forall} A \underset{\text{pol}}{\exists} S_f$: $\mathrm{ADV}_k(S_f^f, f) \geq \left[\mathrm{ADV}_k(A^f, G^f_f)\right]^c$	$\underset{\text{of}}{\forall} \psi \exists \mathcal{F}$: $\forall G \underset{\text{pol}}{\exists} A \underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(A^f, G^f_{\psi(f)})\right] \neq k^{-\omega(1)}$ $\underset{\text{pol}}{\forall} S \underset{\text{of}}{\forall} \varphi \underset{f\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}_k(S^f_{\varphi(f)}, f)\right] = k^{-\omega(1)}$

focuses on (4.1) derives the following averaged version:

$$\underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}^{\mathcal{P}}_k(A, G^f)\right] \neq k^{-\omega(1)} \quad \Rightarrow \quad \underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}^{\mathcal{Q}}_k(S^{f,A}, f)\right] \neq k^{-\omega(1)} \ , \qquad (4.2)$$

and then derives a contradiction based on (S), (B) and (4.2). Indeed, from (S) it follows that $\underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}^{\mathcal{P}}_k(A, G^f)\right] = 1 - k^{-\omega(1)} \neq k^{-\omega(1)}$. By (4.2) we imply that $\underset{f,A\leftarrow\mathcal{F}}{\mathbf{E}} \left[\mathrm{ADV}^{\mathcal{Q}}_k(S^{f,A}, f)\right] \neq k^{-\omega(1)}$ which contradicts (B).

4.3.1 Averaging-Based Separation for Poly-Preserving Reductions

The big limit of the averaging approach described above is that we cannot deduce the averaged condition (4.2) from the general reduction condition (4.1), Specifically, We would like to prove that if $\mathrm{ADV}_k^{\mathcal{Q}}(S^{f,A}, f) = k^{-\omega(1)}$ can imply

$$\mathop{\mathbf{E}}_{f}[\mathrm{ADV}_k^{\mathcal{Q}}(S^{f,A}, f)] = k^{-\omega(1)}$$

According to Lemma 6 (Negligible Average Argument) we know that $\mathrm{ADV}_k^{\mathcal{P}}(A, G^f) = k^{-\omega(1)}$ for measure one of f's, but this does not mean that $\mathop{\mathbf{E}}_{f}[\mathrm{ADV}_k^{\mathcal{P}}(A, G^f)] = k^{-\omega(1)}$ based on Lemma 8. As the guarantee condition is too weak for average-based separation, we use the poly-preserving reductions instead of general reduction condition for the averaging approach.

In the poly-preserving BBB-reductions (fully black-box reductions), the security argument is of the form $\mathrm{ADV}_k^{\mathcal{Q}}(S^{f,A}, f) \geq \big[\mathrm{ADV}_k^{\mathcal{P}}(A, G^f)\big]^c$. For poly-preserving reductions, the averaged reduction condition (4.2) easily follows:

$$\mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\mathrm{ADV}_k^{\mathcal{Q}}(S^{f,A}, f)\big] \geq \mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\big(\mathrm{ADV}_k^{\mathcal{P}}(A, G^f)\big)^c\big] \geq \left(\mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\mathrm{ADV}_k^{\mathcal{P}}(A, G^f)\big]\right)^c ,$$

where the second inequality is an application of the Jensen inequality. This implies that if $\mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\mathrm{ADV}_k^{\mathcal{P}}(A, G^f)\big]$ is non-negligible, then so is $\mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\mathrm{ADV}_k^{\mathcal{Q}}(S^{f,A}, f)\big]$. If $\mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\mathrm{ADV}_k^{\mathcal{Q}}(S^{f,A}, f)\big] = k^{-\omega(1)}$, but at the same time $\mathop{\mathbf{E}}_{f,A\leftarrow\mathcal{F}}\big[\mathrm{ADV}_k^{\mathcal{P}}(A, G^f)\big] \neq k^{-\omega(1)}$, we derive a contradiction.

Based on the original four different types reductions in the averaging-based separation [3], we list another three types of reductions in Table 4.2, All proofs are given in the next section. Note that here we use a stronger reduction condition based on the poly-preserving reductions, hence the breakage condition for averaging-based separation can be somewhat weaker than in the traditional extraction-based approach, we only need require that the success of A is non-negligible.

4.3.2 Proofs for Additional Averaging-Based Separations

First, we give the proof for nonexistence of a BBNa-reductions in averaging-based separation, the idea of the proof originates from [3], only difference is that we use the

nonexistence of BBBa-reduction to show the separation result between two primitives in the leaky-oracle model.

Lemma 9 *The existence of poly-preserving BBNa black-box reductions is equivalent to:*

$$\underset{pol}{\exists} G \underset{pol}{\exists} \mathcal{S} \underset{of}{\exists} \varphi \underset{pol}{\forall} A \forall f\colon \quad \mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f))\right]^c \quad \textit{where } c \geq 1 \ . \tag{4.3}$$

Proof. Assume first that $\underset{\text{pol}}{\exists} G \forall f \underset{\text{pol}}{\exists} S_f \underset{\text{pol}}{\forall} A\colon \ \mathrm{ADV}_k(S^{A,f}_f, f) \geq \left[\mathrm{ADV}_k(A^f, G^f))\right]^c$, i.e. there exists a poly-preserving BBNa black-box reduction and prove (4.3). Let φ be an oracle function so that $\varphi(f)$ is a bit-representation of S_f. Let $\mathcal{S}$ be the universal f-oracle machine, which when given as input a bit-representation $\varphi(f)$ behaves exactly like S_f. This means that $\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f) = \mathrm{ADV}_k(S^{A,f}_f, f)$. Moreover, as such simulation is possible with logarithmic overhead, it follows that $\mathcal{S}_{\varphi(f)}$ is poly-time. As $\mathcal{S}$ and φ are the same for all instances of f, the statement (4.3) follows.

From (4.3) by defining $S_f := \mathcal{S}_{\varphi(f)}$, there exists G such that for all f there is S_f such that for poly-time A, so that $\mathrm{ADV}_k(S^{A,f}_f, f) = \mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f)\right]^c$, which proves the existence of poly-preserving BBNa black-box reduction. □

Theorem 11 *If* $\underset{pol}{\forall} G \exists \mathcal{F} \underset{pol}{\forall} S \underset{of}{\forall} \varphi \underset{pol}{\exists} A\colon$ *so that (I)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(A^f, G^f))\right] \neq k^{-\omega(1)}$ *and (II)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(S^{A,f}_{\varphi(f)}, f)\right] = k^{-\omega(1)}$*, there exists no poly-preserving BBNa reductions.*

Proof. By using (4.3), (I) and (II), we will derive a contradiction. Let G, $\mathcal{S}$ and φ be as in (4.3). By applying the assumption of the theorem to this G, we conclude that there exist a distribution $\mathcal{F}$ with the properties (I) and (II). By applying the assumption of the theorem to the $\mathcal{S}$ and φ, we can get a poly-time oracle machine A with the properties (I) $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(A^f, G^f)\right] \neq k^{-\omega(1)}$(*) and (II) $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\left[\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f)\right] = k^{-\omega(1)}$ (**). Here we use the same A in (4.3) such that $\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f) \geq \left[\mathrm{ADV}_k(A^f, G^f)\right]^c$(***) holds for all f. Finally, by averaging (***) and using the Jensen's inequality we have:

$$\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f)\Big] \geq \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\big[\mathrm{ADV}_k(A^f, G^f)^c\big] \geq \Big[\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\big[\mathrm{ADV}_k(A^f, G^f)\big]\Big]^c \ ,$$

which is a contradiction between (*) and (**). □

Lemma 10 *The existence of poly-preserving NBBa reductions is equivalent to:*

$$\underset{pol}{\exists} S \underset{of}{\exists} \psi \underset{pol}{\exists} \mathcal{P} \underset{pol}{\forall} A \forall f\colon \quad \mathrm{ADV}_k(S^{A,f}, f) \geq \left[\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)})\right]^c \quad \textit{where } c \geq 1 \ . \tag{4.4}$$

Proof. Assume first that $\underset{\text{pol}}{\exists} S \forall f \underset{\text{pol}}{\exists} G_f \underset{\text{pol}}{\forall} A\colon \ \mathrm{ADV}_k(S^{A,f}, f) \ \geq \ \Big[\mathrm{ADV}_k(A^f, G_f^f)\Big]^c$, i.e. there exists a poly-preserving NBBa reduction, and prove (4.4). Let ψ be a mapping so that $\psi(f)$ is the bit-string representation of G_f. Let $\mathcal{P}$ be the universal f-oracle machine so that $\mathcal{P}_{\psi(f)}$ behaves identical to G_f. Hence, $\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)}) = \mathrm{ADV}_k(A^f, G^f_f)$, and due to the efficiency of simulation, $\mathcal{P}_{\psi(f)}$ is poly-time. As $\mathcal{P}$ and ψ are the same for all instances of f, the statement (4.4) follows. □

Theorem 12 *If* $\underset{pol}{\forall} S \underset{of}{\forall} \psi \exists \mathcal{F} \underset{pol}{\forall} G \underset{pol}{\exists} A\colon$ *so that (I)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(A^f, G^f_{\psi(f)})\Big] \neq k^{-\omega(1)}$ *and (II)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\big[\mathrm{ADV}_k(S^{A,f}, f)\big] = k^{-\omega(1)}$*, there exists no poly-preserving NBBa reductions.*

Proof. By using (4.4), (I) and (II), we derive a contradiction. Let ψ, $\mathcal{P}$ and S be as in (4.4). By applying the assumption of the theorem to S and ψ, we conclude that a distribution $\mathcal{F}$ with the properties (I) and (II). By applying the assumption of the theorem to $\mathcal{P}$, we conclude that there exists A such that $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)})\Big] \neq k^{-\omega(1)}$(*) and $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\big[\mathrm{ADV}_k(S^{A,f}, f)\big] = k^{-\omega(1)}$(**). Here we use the same A in (4.4) such that $\mathrm{ADV}_k(S^{A,f}, f) \geq \Big[\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)})\Big]^c$(***) holds for all f. Finally, by averaging (***) and using the Jensen's inequality, we have

$$\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\big[\mathrm{ADV}_k(S^{A,f}, f)\big] \geq \underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)})^c\Big] \geq \Big[\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)})\Big]\Big]^c \ .$$

A contradiction between (*) and (**). □

Finally, we give the proof for nonexistence of poly-preserving NBNa reductions, this proof can consider as a combination of the result from previous two proofs.

Lemma 11 *The existence of poly-preserving NBNa reductions is equivalent to:*

$$\underset{of}{\exists}\psi \underset{pol}{\exists}\mathcal{P} \underset{pol}{\exists}\mathcal{S} \underset{of}{\exists}\varphi \underset{pol}{\forall} A \forall f\colon \quad \mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)}, f) \geq \Big[\mathrm{ADV}_k(A^f, \mathcal{P}^f_{\psi(f)})\Big]^c \quad \textit{where } c \geq 1 \ . \tag{4.5}$$

Proof. Assume first that $\forall f \underset{\text{pol}}{\exists} G_f \underset{\text{pol}}{\exists} S_f \underset{\text{pol}}{\forall} A\colon \ \mathrm{ADV}_k(S^{A,f}_f, f) \ \geq \ \Big[\mathrm{ADV}_k(A^f, G^f_f)\Big]^c$, i.e. there exists a poly-preserving NBNa black-box reduction, and prove (4.5). Here we define ψ and $\mathcal{P}$ like in Lemma 4.4, for $\mathcal{S}$ and φ like in Lemma 4.3, As $\mathcal{S}$, φ, $\mathcal{P}$ and ψ are the same for all instances of f. The statement (4.5) follows. □

Theorem 13 *If* $\underset{of}{\forall}\psi\exists\mathcal{F}\underset{pol}{\forall}G\underset{pol}{\forall}S\underset{of}{\forall}\varphi\underset{pol}{\exists}A\colon$ *so that (I)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(A^f, G^f_{\psi(f)})\Big] \neq k^{-\omega(1)}$ *and (II)* $\underset{f\leftarrow\mathcal{F}}{\mathbf{E}}\Big[\mathrm{ADV}_k(S^{A,f}_{\varphi(f)}, f)\Big] = k^{-\omega(1)}$*, there exists no poly-preserving NBNa*

reductions.

Proof. By using (4.5), (I) and (II), we derive a contradiction. Let ψ, $\mathcal{P}$, $\mathcal{S}$ and φ be as in (4.5). By applying the assumption of the theorem to ψ, we conclude that a distribution $\mathcal{F}$ with the properties (I) and (II). By applying the assumption of the theorem to $\mathcal{P}$, $\mathcal{S}$ and φ we conclude that there exists A such that $\mathop{\mathbf{E}}_{f\leftarrow\mathcal{F}}\left[\mathrm{ADV}_k(A^f,\mathcal{P}^f_{\psi(f)})\right] \neq k^{-\omega(1)}$(*) and $\mathop{\mathbf{E}}_{f\leftarrow\mathcal{F}}\left[\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)},f)\right] = k^{-\omega(1)}$(**). Here we use the same A in (4.5) such that $\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)},f) \geq \left[\mathrm{ADV}_k(A^f,\mathcal{P}^f_{\psi(f)})\right]^c$(***) holds for all f. Finally, by averaging (***) and using the Jensen's inequality, we have

$$\mathop{\mathbf{E}}_{f\leftarrow\mathcal{F}}\left[\mathrm{ADV}_k(\mathcal{S}^{A,f}_{\varphi(f)},f)\right] \geq \mathop{\mathbf{E}}_{f\leftarrow\mathcal{F}}\left[\mathrm{ADV}_k(A^f,\mathcal{P}^f_{\psi(f)})^c\right] \geq \left[\mathop{\mathbf{E}}_{f\leftarrow\mathcal{F}}\left[\mathrm{ADV}_k(A^f,\mathcal{P}^f_{\psi(f)})\right]\right]^c .$$

A contradiction between (*) and (**). □

5 Future Research

In this thesis we mainly considered the reduction and separation results from semi level above. Because Reingold et al [9] prove that if one-way functions imply key agreement via a free reduction, then there is also a weakly-BB reduction from key agreement to one-way functions.

Clearly, from the Definition 12 we know that black-box separations result are presumably impossible in free reductions. The proof by Reingold et al. shows that free reductions and weakly-reduction are basically equivalent, hence black-box separations results are also impossible in weakly-reductions. However, Pass et al. [26] gave a new proof to show that one-way functions do not imply one-way-permutations through a meta-reduction based on a $\forall\exists$-weakly-BB reduction. Obviously this is a contradiction from Reingold's result, and let us think following questions: does the proof for equivalence of weakly reduction and free reduction hold for all cases. If we can get separations result in weakly-reductions, what will be the relation between weakly-reductions and other reductions, here we give a conjecture as follows.

XNNa and weakly-XNNa Reductions are equivalent, where X$\in\{$B,N$\}$. From the definitions we know the only difference between XNNa and weakly-XNNa-reductions is whether the adversary A can get access to an instance f or not. If we consider the same approach used to prove the equivalence of BNNa and BBNa, we notice that we might still fix the restrictions and derive a contradiction from the assumption, and it seems like this approach still works for the proof of XNN and weakly-XNN reductions.

It would also be interesting to know the relations between different types of poly-preserving reductions. In this thesis we mainly considered the relation between different notions of ordinary reductions, but for poly-preserving reductions, we only use them to prove the averaging-based separation. As a stronger version of reductions, our result for relation between different notions of reductions maybe totally different for poly-preserving reductions. Here we also give a conjecture about poly-preserving reductions as follows.

XBZa and XNZa Poly-Preserving Reductions are Inequivalent, where X,Z$\in\{$B,N$\}$. We know that for poly-preserving reductions the guarantee condition was strengthened as $\textsc{Adv}_k^{\mathrm{Q}}(S^f, f) \geq \left[\textsc{Adv}_k^{\mathrm{P}}(\mathsf{A}^f, G^f)\right]^c$, here if we consider the same approach used to prove the equivalence of BNNa and BBNa, we can find a fixed adversaries A that satisfy the whole statement. But as this guarantee condition is an inequality, we need consider the whole statement including a different A, and it seems like we cannot derive a contradiction base on this inequality.

6 Conclusions

In this thesis, we achieved two goals which are strongly related to cryptographic reductions and separations. First, we further consider the relations between different types of reductions, we get some new results for efficient adversaries in CAP framework in Chapter 3. Second, based on the work in [3], we complement all notions of black-box reductions for efficient adversaries in the traditional oracle extraction-based separation and the averaging-based separation according to the CAP framework in Chapter 4.

We believe that the study of relationships among different notions of reductions and oracle separation techniques are important. Since separation techniques critically depend on the types of reductions, and separation results can help cryptographers to make clear fundamental differences between primitives. They can also save a lot of wasted effort by guiding researchers away from hopeless approaches. In this thesis, we reduce the original CAP framework for efficient adversaries into a simpler and easier to understand way. We also complement all notions of black-box reductions in both the traditional oracle extraction-based separation and the averaging-based separation, which can help researchers to get separation results more quickly. For these reasons, we believe that our result is valuable in the study of cryptographic reductions and separations.

Many open questions remain for our following work, both in the relation between different types of reductions and in averaging-based separation. Some such questions include: the relations between weakly black-box-reduction and other types of black-box-reductions, the relations between different types of reductions based on poly-preserving reductions. Is it possible to get averaging-based separation below BNBa reductions for poly-preserving reductions and how to confirm the result of the relation between BBBa and BNBa reductions.

References

[1] Buldas, A., Jürgenson, A., Niitsoo, M.: Efficiency bounds for adversary constructions in black-box reductions. In: C. Boyd and J. Gonz.alez Nieto (Eds.): ACISP 2009, LNCS 5594, pp. 264–275 (2009)

[2] Buldas, A., Laur, S., Niitsoo, M.: Oracle separation in the non-uniform model. In: J.Pieprzyk and F.Zhang (Eds.): ProvSec 2009, LNCS 5848, pp. 230-244 (2009)

[3] Buldas, A., Laur, S., Niitsoo, M. Black-box separations and their adaptability to the non-uniform model. In: C. Boyd and L. Simpson (Eds.): ACISP 2013, LNCS 7959, pp. 152-167 (2013)

[4] Gennaro, R., Gertner, Y., Katz, J.: Lower bounds on the efficiency of encryption and digital signature schemes. In: STOC 2003, pp. 417–425 (2003)

[5] Gennaro, R., Gertner, Y., Katz, J., Trevisan, L.: Bounds on the efficiency of generic cryptographic constructions. SIAM journal on Computing, 35, 217–246 (2006)

[6] Gertner, Y., Kannan, S., Malkin, T., Reingold, O., Viswanathan, M.: The relationship between public key encryption and oblivious transfer. In: FOCS'00. pp. 325–335 (2000)

[7] Hsiao, C.Y., Reyzin, L.: Finding collisions on a public road, or do secure hash functions need secret coins? In: Franklin, M. (Ed.): CRYPTO 2004, LNCS 3152, pp. 92–105 (2004)

[8] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of oneway permutations. In Proceedings of the 21st Annual ACM Symposium on Theory of Computing, pages 44–61 (1989)

[9] Reingold, O., Trevisan, L., Vadhan, S.: Notions of reducibility between cryptographic primitives. In: Naor, M. (Ed.): TCC04, LNCS 2951, pp. 1–20 (2004)

[10] Simon, D.: Finding collisions on a one-way street: can secure hash functions be based on general assumptions? In: Nyberg, K. (Ed.): Eurocrypt'98, LNCS 1403, pp. 334–345 (1998)

[11] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby: A pseudorandom generator from any one-way function. SIAM Journal on Computing, 28(4):1364-1396 (1999)

[12] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. Journal of the ACM, 33:792-807 (1986)

[13] Moni Naor. Bit commitment using pseudorandomness. Journal of Cryptology, 4(2):151–158 (1991)

[14] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. Journal of the ACM, 38(1):691–729 (1991)

[15] Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. 28(2), 270–299 (1984)

[16] Baecher, P., Brzuska, C., Fischlin, M.: Notions of black-box reductions, revisited. IACR Cryptology ePrint Archive (2013)

[17] Steven Rudich. Limits on the provable consequences of one-way functions. PhD thesis, U.C. Berkeley (1988)

[18] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In Proceedings of the IEEE Symposium on Foundations of Computer Science (2000)

[19] Dan Boneh and Ramarathnam Venkatesan. Breaking RSA may not be equivalent to factoring. In Kaisa Nyberg, editor, EUROCRYPT'98, volume 1403 of LNCS, pages 59-71. Springer (1998)

[20] Fischlin, M., Lehmann, A., Ristenpart, T., Shrimpton, T., Stam, M., Tessaro, S.: Random Oracles with(out) Programmability. In: Abe, M. (ed.) ASIACRYPT 2010.

LNCS, vol. 6477, pp. 303–320. Springer, Heidelberg (2010)

[21] Dodis, Y., Oliveira, R., Pietrzak, K.: On the Generic Insecurity of the Full Domain Hash. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 449–466. Springer, Heidelberg (2005)

[22] Boldyreva, A., Cash, D., Fischlin, M., Warinschi, B.: Foundations of Non-malleable Hash and One-Way Functions. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 524–541. Springer, Heidelberg (2009)

[23] Marc Fischlin and Dominique Schröder. On the impossibility of three-move blind signature schemes. In Henri Gilbert, editor, EUROCRYPT 2010, volume 6110 of LNCS, pages 197-215. Springer (2010)

[24] Emmanuel Bresson, Jean Monnerat, and Damien Vergnaud. Separation results on the one-more" computational problems. In Tal Malkin, editor, CT-RSA 2008, volume 4964 of LNCS, pp. 71-87. Springer (2008)

[25] Rafael Pass. Limits of provable security from standard assumptions. In Lance Fortnow and Salil P. Vadhan, editors, 43rd Annual ACM Symposium on Theory of Computing, pp. 109–118, San Jose, California, USA, June 6–8, ACM Press (2011)

[26] Rafael Pass, Wei-Lung Dustin Tseng, and M. Venkitasubramaniam. Towards non-black-box lower bounds in cryptography. In Yuval Ishai, editor, volume 6597 of LNCS, pp. 579-596. Springer (2011)

[27] Luby, M.: Pseudorandomness and cryptographic applications. pp.31-34. Princeton University Press, Princeton (1996)