DOCTORAL THESIS

Advanced Modelling Frameworks for the Digital Twin of an Autonomous Electric Vehicle Propulsion Drive System

Mahmoud Ibrahim Hassanin Mohamed

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY
TALLINN 2025

TALLINN UNIVERSITY OF TECHNOLOGY
DOCTORAL THESIS
31/2025

Advanced Modelling Frameworks for the Digital Twin of an Autonomous Electric Vehicle Propulsion Drive System

MAHMOUD IBRAHIM HASSANIN MOHAMED

TALTECH PRESS

TALLINN UNIVERSITY OF TECHNOLOGY
School of Engineering
Department of Electrical Power Engineering and Mechatronics
This dissertation was accepted for the defense of the degree 09/04/2025

Supervisor:	Prof. Anton Rassõlkin School of Engineering Tallinn University of Technology Tallinn, Estonia
Opponents:	Assoc. Prof. Lassi Aarniovuori Electrical Engineering LUT School of Energy Systems Lappeenranta, Finland
	Assist. Prof. Ramy Georgious Electrical and Electronic Engineering Department University of Oviedo Asturias, Spain

Defence of the thesis: 09/06/2025, Tallinn

Declaration:
Hereby I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology has not been submitted for doctoral or equivalent academic degree.

Mahmoud Ibrahim

----
signature

ISSN 2585-6898 (publication)
ISBN 978-9916-80-293-9 (publication)
ISSN 2585-6901 (PDF)
ISBN 978-9916-80-294-6 (PDF)
DOI https://doi.org/10.23658/taltech.31/2025
Printed by Koopia Niini & Rauam

Ibrahim, M. (2025). *Advanced Modelling Frameworks for the Digital Twin of an Autonomous Electric Vehicle Propulsion System* [TalTech Press]. https://doi.org/10.23658/taltech.31/2025

TALLINNA TEHNIKAÜLIKOOL
DOKTORITÖÖ
31/2025

Autonoomse elektrisõiduki veoelektriajami digitaalse kaksiku täiustatud modelleerimise raamistikud

MAHMOUD IBRAHIM HASSANIN MOHAMED

TALTECH KIRJASTUS

Contents

List of Publications

The list of author's publications, based on which the thesis has been prepared:

I **Ibrahim, M**.; Rassõlkin, A.; Vaimann, T.; Kallaste, A. Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System. Sustainability 2022, 14, 601. https://doi.org/10.3390/su14020601.

II **Ibrahim, M**.; Rjabtšikov, V.; Gilbert Zequera, R. A. Overview of Digital Twin Platforms for EV Applications. Sensors 2023, 23, 1414. https://doi.org/10.3390/s23031414.

III **M. Ibrahim**, V. Rjabtšikov, A. Rassõlkin, T. Vaimann and A. Kallaste, "Validation of an EV-Permanent Magnet Synchronous Motor Model Based on Analytical Dynamic Approach," 2022 International Conference on Electrical Machines (ICEM), Valencia, Spain, 2022, pp. 2384–2390, doi: 10.1109/ICEM51905.2022.9910755.

IV **M. Ibrahim**, V. Rjabtšikov, S. Jegorov, A. Rassõlkin, T. Vaimann and A. Kallaste, "Conceptual Modelling of an EV-Permanent Magnet Synchronous Motor Digital Twin," 2022 IEEE 20th International Power Electronics and Motion Control Conference (PEMC), Brasov, Romania, 2022, pp. 156–160, doi: 10.1109/PEMC51159.2022.9962943.

V **Ibrahim, M.**, Rjabtšikov, V. and Rassõlkin, A. (2025), Digital shadow of an electric vehicle-permanent magnet synchronous motor drive for real-time performance monitoring. Dig Twins and App e12024. https://doi.org/10.1049/dgt2.12024

VI **M. Ibrahim**, H. A. Raja, A. Rassõlkin, T. Vaimann and A. Kallaste, "An EV-Traction Inverter Data-Driven Modelling for Digital Twin Development," 2023 23rd International Scientific Conference on Electric Power Engineering (EPE), Brno, Czech Republic, 2023, pp. 1–5, doi: 10.1109/EPE58302.2023.10149230.

VII **M. Ibrahim,** A. Rassõlkin and V. Rjabtšikov, "Reverse Engineering-Based Modeling of an EV Motor Drive for Digital Twin Development," 2024 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference (ESARS-ITEC), Naples, Italy, 2024, pp. 1–5, doi: 10.1109/ESARS-ITEC60450.2024.10819839.

Author's Contribution to the Publications

Contribution to the papers in this thesis are:

I Mahmoud Ibrahim is the primary author of this article. He conducted an extensive literature review on Digital Twins and their applications including Electric Vehicles. He also did a comparative review of different technologies starting with computer simulations and ending with Digital Twins

II Mahmoud Ibrahim is the primary author of this article. He investigated different platforms used for Digital Twin development. He wrote the draft of the paper.

III Mahmoud Ibrahim is the first author of this article. He prepared the full draft, and the models used in it. He also collected the lab data used to predict the motor parameters.

IV Mahmoud Ibrahim is the primary author of this article. He prepared the primary model of the Permanent Magnet Synchronous Motor using a physics-based technique. He also functioned the model to receive data from the physical system in real-time. He wrote the draft of the paper.

V Mahmoud Ibrahim is the primary author of the article. He wrote the paper draft. He developed the data-driven technique based on the Artificial Neural Network structure to model the performance of the traction inverter.

VI Mahmoud Ibrahim is the first Author of the article. He developed and functioned the Digital Twin model to serve as a real-time soft sensor for the vehicle's torque and speed. He also prepared the paper draft.

VII Mahmoud Ibrahim is the first author of the article. He wrote the paper draft. He used the data-driven surrogate modeling technique to build the digital model of the EV propulsion drive system.

1 Introduction and Motivation

The rapid advancement in electric vehicle (EV) technology has been a defining feature of the automotive industry in recent years. This shift towards electrification is driven by the pressing need to reduce greenhouse gas emissions, decrease dependency on fossil fuels, and mitigate the impact of climate change. EVs provide various advantages, including lower operating costs, reduced maintenance needs, and improved energy efficiency. As a result, they are becoming an increasingly viable and attractive option for consumers and businesses worldwide. EVs, encompassing fully electric (battery) and hybrid vehicles, have continued to experience significant growth. According to an EV volumes survey, by 2024, the global market will see a substantial increase, with EV sales reaching new highs compared to previous years. Global volumes grew 35% year-on-year in 2024 to reach 14.2 million units, equating to a market share of 16.7%, up from 13.6% in 2023. Looking ahead, EV Volumes expects 16.6 million EV sales in 2025, equating to a 19.2% light-vehicle market share. According to the latest projections by Information Handling Services (IHS) Markit, EVs are expected to capture 45% of the new car global market by 2040 and nearly 80% by 2050 [1]. Figure 1 illustrates a growth comparison between battery electric vehicles (BEVs), plug-in hybrid electric vehicles (HEVs), fuel cell vehicles (FCVs), and traditional internal combustion engine (ICE) vehicles as inspired from [2].

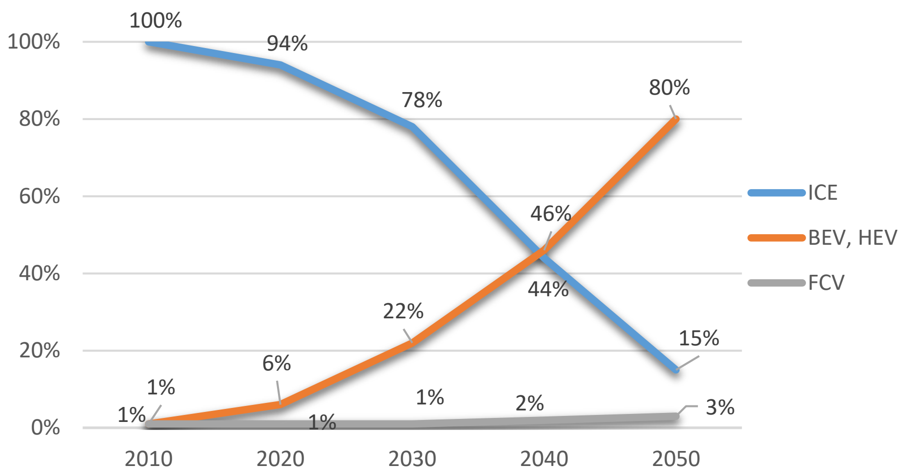


Figure 1. Growth comparison of the global battery electric vehicle (BEV), internal combustion engine (ICE) vehicles, and fuel cell vehicle (FCV) markets.

This growth in the EV industry is driven by significant technological advancements, reduced manufacturing costs, and supportive international policies, as is the case with the EU vision. The EU directive to cease the sale of new internal combustion engine (ICE) vehicles by 2035 emphasizes the EU's dedication to environmental sustainability and encourages innovation in the automotive sector. This legislation has driven manufacturers to focus on EV technology, resulting in rapid advancements in charging infrastructure, battery technology, and vehicle efficiency. Besides promoting EV development, the ban supports consumer adoption of electric mobility, leading to a market shift towards cleaner transportation options. Additionally, the emergence of autonomous vehicles highlights the integration of advanced technologies in transportation. EVs incorporate automation and electrification, utilizing sensors, cameras, and sophisticated algorithms to navigate complex traffic scenarios autonomously [3].

The role of simulation is essential to this progress, which has revolutionized how EV systems are designed, tested, and optimized. Initially, simulations in the EV sector were grounded in basic mathematical models and analytical methods, constrained by the limited computational power available. These early simulations provided a foundation for understanding fundamental aspects of EV performance and battery management but were often limited in scope and accuracy. Over time, more sophisticated computational techniques and high-performance computing (HPC) integration have transformed simulation capabilities, allowing for detailed analysis and optimization of complex systems. As the automotive industry advances toward the next generation of vehicles, particularly Software-Defined EVs (SDEVs), the importance of simulations has become even more pronounced. Advanced simulations are crucial for developing and validating the complex software that defines vehicle behavior, enabling seamless integration of features like autonomous driving and intelligent energy management. The progression from early simulation methods to advanced techniques like Hardware-in-the-Loop (HiL) simulation has been instrumental in bridging the gap between virtual models and physical systems, enabling real-time testing and validation [4]. This historical development of simulation technology has set the stage for the emergence of Digital Twins (DT), which represents the next frontier in engineering and manufacturing innovation, as the following subsections illustrate.

1.1 From Computer Simulations to Digital Twins

The history of simulation dates to World War II when mathematicians Jon Von Neumann and Stanislaw Ulam encountered challenges with neutron behavior. Traditional trial-and-error methods were too expensive, so they proposed the roulette wheel method [5]. This involved using known basic data on event occurrences and merging the probabilities of individual events in a step-by-step analysis to predict the outcome of an entire sequence of events. Their technique successfully solved the neutron problem and quickly gained popularity, finding applications in industrial contexts. In [6] Shannon described simulation as “the process of designing the operation of a system.” Using computers in simulation involves representing the dynamic responses of a system by modeling another system after it. A simulation uses a mathematical model of a real system in the form of a computer program. This model comprises equations that replicate the functional relationships within the physical system. When the program runs, the resulting mathematical dynamics form an analog of the real system’s behavior, with the results presented as data. For example, an electric machine can be modeled mathematically with variables such as current, voltage, and magnetic flux, and additional equations can adjust for changes in variables like winding material to define heat dissipation losses. Simulations can also be computer graphics images representing dynamic processes in animated sequences. However, this technology has some drawbacks, such as programming errors, the time required to interpret results, no data exchange between the real and simulation models, and potentially unrealistic or unreliable human reactions to the model [7].

As computing power increased through the 1980s and 1990s, so did the complexity and accuracy of simulations. Figure 2 illustrates the HiL system. The introduction of hardware-in-the-loop (HiL) simulations marked a significant milestone. HiL simulation is a technique where physical signals from a controller are connected to a test system that simulates real-world conditions, effectively deceiving the controller into operating as if it were in the actual product. This allows for iterative testing and design as though the physical system is being used, enabling the execution of thousands of potential scenarios without the costs and time associated with physical tests [8].

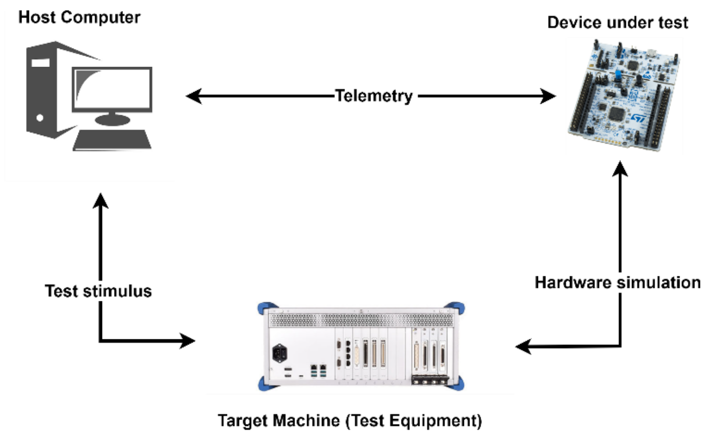


Figure 2. Hardware-in-Loop (HiL) system.

HiL simulation is particularly valuable for testing control algorithms without needing a physical system, especially when testing on a real system is expensive or dangerous [9]. HiL simulation is extensively used in the automotive, aerospace, defense, and industrial automation sectors to test embedded designs. It is also gaining traction in medical devices, communications, semiconductors, and other industries. The effectiveness of HiL relies heavily on the quality of the simulation software, which must be paired with hardware that accommodates system specifications such as connector types and I/O capabilities and allows for fault insertion and real-world scenario testing [10] An EV motor control unit (MCU) is a straightforward example of a HiL system. The MCU converts sensor data into actions, like adjusting the inverter frequency when the accelerator is pressed. In the HiL test, the physical motor is replaced with a simulation that uses hardware and software to interact with real I/O as if the motor were present. This allows rapid software updates, broad scenario testing, and comprehensive coverage without risking the physical system. Despite the slightly higher cost of real-time simulators, the main challenges of HiL systems are the complexity of development and verification.

DT has emerged to overcome the complexities and limitations of HiL systems and meet the growing demands for higher performance and operational self-awareness, DT has emerged. The concept of DT has been present since the early 2000s. Michael Grieves first introduced a DT model in 2002. In 2012, NASA released a seminal paper defining DTs as integrated, Multiphysics, multiscale, and probabilistic systems simulations. These simulations use the best physical models, sensors, and historical data to replicate the lifecycle of their physical counterparts [11]. Chen and Huang [12] described DTs as computerized models that mirror the functional features of physical systems. Zheng et al. [13] defined DTs as virtual information sets representing physical assets. Mandi [14] describes DTs as virtual instances of physical systems that are continually updated with performance, maintenance, and health status data throughout the physical system's lifecycle. DTs, or computational mega models, are digital representations of physical objects, processes, or services. These can range from small items like jet engines to large structures such as buildings or entire cities. Beyond physical assets, DT technology can replicate processes to collect data and predict performance [15]. The following Figure 3 proposes a timeline of simulation technology evolution.

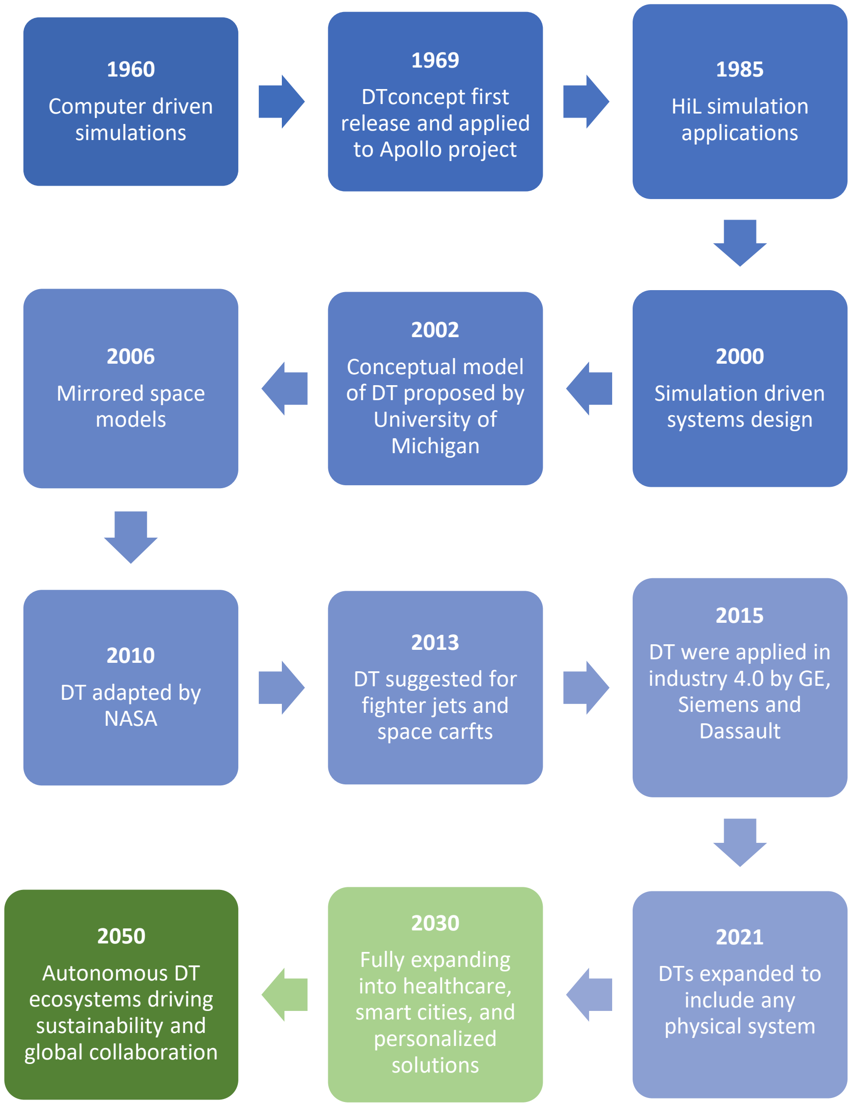


Figure 3. Timeline of simulation technology evolution.

A DT is essentially an asset of computer programs that uses real-world data to create simulations predicting the performance of products or processes. These programs can enhance their outputs by incorporating the Internet of Things (IoT), Artificial Intelligence (AI), and software analytics. With advances in machine learning and the availability of Big Data, DTs have become fundamental in modern engineering, driving innovation and improving performance.

At first glance, DT and HiL simulations appear similar as both involve real-time simulations. However, the key distinction lies in their approach. HiL involves building a software model of the core system that interfaces with and controls real hardware components (such as circuits and mechanical parts) to evaluate the controller's performance. In contrast, DT involves creating an entirely software-based model of the entire system, which interacts with the controller's inputs and outputs to assess how well the controller performs its intended functions. The comparison of DT and HiL is summarized in Table 1 [9].

Table 1. Comparison between Hardware-in-Loop (HiL) and Digital Twin (DT).

Points of Comparison	**Hardware-in-Loop (HiL)**	**Digital Twin (DT)**
Simulation type	Real-time	Real-time
Functions	Design; Testing; Optimization; Fault Detection	Design; Diagnosis; Optimization; Predictive Maintenance; Fault Detection; Health Monitoring; Lifetime Prediction
Real-Time Data	Limited use of real-time data, primarily for controller testing.	Designed for continuous two-way flow of real-time data.
Scope	Component, Subsystem	Component, Asset, System, Process
Flexibility	Less flexible due to hardware constraints	Highly flexible, allowing for rapid modifications and testing of various scenarios

Although simulations and DTs utilize digital models to replicate a system's processes, a DT offers a considerably richer virtual environment for analysis. The primary difference between a DT and a conventional simulation is largely a matter of scale. While a simulation typically focuses on a single process, a DT can run multiple simulations to study various processes simultaneously. Moreover, simulations usually do not incorporate real-time data, whereas DTs are designed for continuous two-way information flow. This involves sensors on the physical object providing relevant data to the system processor, and the insights generated by the processor are then shared back with the original object. With access to better and constantly updated data from a wide range of sources, combined with the enhanced computing power of a virtual environment, DTs can analyze more issues from multiple perspectives than standard simulations. This greater analytical capability significantly enhances the potential to improve products and processes.

1.2 Digital Twin Structure

The structure of a DT is multi-layered, beginning with a physical system, which can range from a single component to an entire system or process. This physical layer has sensors and IoT devices that continuously collect data on various parameters and operational performance. The next layer is the communication and data acquisition layer, responsible for the secure and efficient data transmission from the physical entity to the digital realm. This involves protocols and technologies like Message Queuing Telemetry Transport (MQTT), Hypertext Transfer Protocol (HTTP), Controlled Area Network (CAN), and 5G to ensure seamless connectivity and data flow. Once the data reaches the virtual model, the core of the DT, it is processed and stored in a data management system. This system uses advanced analytics, machine learning algorithms, and big data techniques to analyze the incoming data, extract meaningful insights, and predict future behaviors. The virtual model is a high-fidelity replica of the physical entity, accurately simulating its behavior and performance under various conditions. Another crucial component is the integration layer, which connects DT with other enterprise systems such as ERP, PLM, and CRM. This integration enables a holistic view of the operation, facilitating better decision-making and coordination across different departments. The user interface layer allows interaction with the DT through dashboards, 3D visualizations, and augmented reality interfaces. This layer provides an intuitive and user-friendly way to visualize the physical and virtual states, conduct simulations, and derive actionable insights. Finally, the feedback loop ensures that insights and recommendations generated by the DT are relayed back to the physical entity. This loop enables continuous improvement and optimization, allowing for proactive maintenance, improved efficiency, and reduced downtime[16]. Figure 4 provides a detailed illustration of the multilayer DT structure.

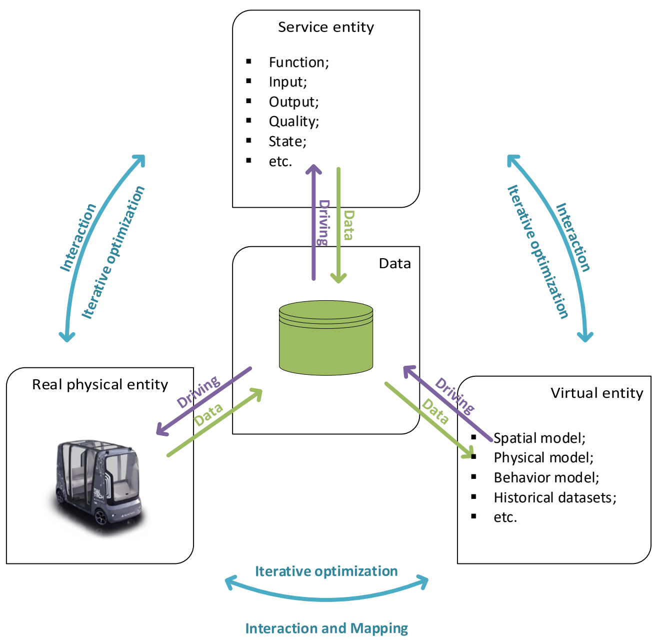


Figure 4. Multilayer Digital Twin (DT) structure [16]

1.3 Digital Twins in Electric Vehicles (EV) Applications

As the fourth Industrial Revolution advances, EV manufacturers increasingly incorporate technology to enhance their production processes and make operations more cost-effective. Advanced machine learning tools and optimization algorithms have significantly contributed to the development of EVs. IoT and DT technologies provide the essential framework for mapping physical assets to digital models. Given that EVs generate substantial sensory data, DT technology surpasses other methods like HiL simulations. The potential and motivations to use DT technology in EVs are immense. DTs can significantly improve several areas, including enhanced design and testing processes, optimized energy consumption, and extended lifetime. DTs can help manufacturers identify potential issues and refine vehicle designs by simulating various driving conditions and user behaviors before building physical prototypes. This reduces development costs and accelerates the time-to-market for new EV models [17]. Furthermore, DTs enable real-time monitoring and analysis of EV performance, allowing for immediate adjustments and improvements. This proactive approach helps minimize downtime and enhance overall vehicle reliability. Integrating EV battery management systems, electric powertrains, and advanced software controls requires high-fidelity simulations to ensure accuracy and reliability. However, this integration also demands substantial computational power, particularly when adopting DTs, necessitating high computational speeds for real-time data processing applications. This can be challenging, as high-fidelity simulation models are often slower and require significant computational resources.

This is the main issue with DT implementation in EV technology: balancing the need for detailed, accurate simulations with the requirement for real-time performance and processing capabilities. Achieving this balance is crucial for leveraging DT's full potential in enhancing EV design, development, and maintenance.

1.4 Thesis Objectives

- Introducing DT technology to older generations of EVs, which were manufactured before the advent of SDEVs and modern DT systems. This can improve operational efficiency, anticipate maintenance needs, and extend the useful life of these older EVs, thus contributing to their sustainability and reliability in current applications.
- Define and establish a comprehensive modelling framework for developing DT of a preexisting repurposed EV propulsion drive system, serving as a detailed guide for researchers and developers to replicate the system performance mitigating its complexities accurately.
- Incorporate robust model identification techniques to handle partially unknown physical model parameters, ensuring precise representation of each component and enhancing overall model reliability.
- Leverage model-based (physics-based) approaches for components with well-defined parameters to replicate them in the digital world.
- Apply data-driven modeling techniques to components where physics-based models face limitations due to complexities or unknowns, enhancing overall model accuracy.
- Implement reduced-order modeling techniques (Hybrid Modeling) to improve computational speed without significantly sacrificing accuracy, making the models suitable for real-time applications, particularly for the powertrain.
- Conduct comprehensive validation against experimental and real-world data, demonstrating the models' ability to predict system behavior under various operating conditions with a high level of accuracy.
- Deploy the developed DT models in real-world or simulated environments to evaluate their performance, integration, and benefits in actual operation, ensuring the framework's effectiveness and versatility across various applications.

1.5 Scope and Challenges

The scope of this thesis encompasses developing and implementing the DT modelling framework for a preexisting electric powertrain repurposed for a new EV application.

1.5.1 Scope

The scope of this thesis centers on developing a comprehensive DT modelling framework of the EV propulsion system, with a specific focus on the electric motor, inverter, and control systems. The modeling process begins with gathering and collecting all necessary data of the physical system to build accurate models of these core components. Different modeling techniques and platforms are explored to create precise virtual representations that capture the dynamics and behavior of the physical system.

1.5.2 Challenges

- **Data Availability and Parameter Uncertainty** related issues. Some propulsion system components suffer from parameter uncertainty due to insufficient or imprecise data. This lack of precise information necessitates using advanced modeling techniques, which require considerable time and effort to develop and validate.
- **System Complexity** of the studied EV propulsion system, which is composed of various components, each requiring distinct modeling approaches. Integrating these diverse models into a unified environment demands seamless interaction between different platforms, a technically challenging process that adds complexity to the project.
- Combining different models and platforms into a comprehensive DT model requires **significant computational resources**, especially for real-time simulations. The heavy computational burden can hinder DT's ability to perform real-time tasks such as predictive maintenance or system optimization, limiting its practical effectiveness.
- Accurately incorporating environmental factors – which can significantly impact the propulsion system's performance – into the DT framework is complex. Modeling these factors adds substantial computational and technical challenges, increasing the overall burden on system design and resources. However, **environmental factors modeling is not part of** the scope of this thesis.
- Achieving **interoperability challenges** between the DT and existing systems and technologies is a critical hurdle. Many of these systems operate on different standards and protocols. Ensuring seamless integration across diverse platforms is especially challenging in multi-component systems like EV propulsion systems.

1.6 Scientific Contributions

1.6.1 Scientific Novelty

- Adapting DT technology to preexisting EV powertrains produced before the era of connected vehicles, addressing the challenges posed by limited interconnectivity and the absence of modern DT frameworks.
- Systematic methodology development to handle and mitigate the impact of uncertainties and partially unknown parameters inherent in physical systems, ensuring accurate representation and reliable predictions.

- Providing a guidance framework for researchers to identify and implement the most appropriate modeling techniques based on the availability and nature of physical system data, serving as a valuable resource for DT modelling procedures.
- Proposing a flexible validation approach that integrates experimental data with correlated real-world measurements, ensuring robust model assessment even without specific sensors.
- Creating a DT framework that integrates physics-based (model-based) and data-driven approaches, forming hybrid models for EV propulsion systems. This framework addresses the individual limitations of each approach when applied in isolation, offering a balanced and effective modeling solution.
- Demonstrating the deployment of the hybrid DT model on a real-time target machine, rigorously testing its performance under realistic operating conditions. This highlights the model's capability for real-time applications, including condition monitoring, predictive maintenance, and operational optimization.

1.6.2 Practical novelty

- Adaption of DT technology in repurposed second-life propulsion systems can improve operational efficiency, anticipate maintenance needs, and extend useful life, which contributes to sustainability and waste reduction.
- Enhancement of System Performance: By operating in real-time, the DT models enable continuous monitoring and optimization of the propulsion system. This improves performance metrics such as efficiency, responsiveness, and reliability.
- Scalable DT modelling framework that is adaptable and suitable across different systems, showcasing their versatility and potential for wide-ranging EV propulsion systems.

1.7 Thesis Structure

The thesis is organized into six chapters as follows:

- Chapter 1: Introduction

This chapter introduces the thesis topic, outlines the research problem, and identifies the main research objectives addressed by the thesis.

- Chapter 2: Literature Review

This chapter provides an overview of recent works on the topic, critically analyzing existing studies and identifying areas for further research.

- Chapter 3: Physical System Overview

This chapter details the specific system under study and its core components. It provides a comprehensive description of the system architecture and its operational principles.

- Chapter 4: Motor Digital Twin Modelling Framework

This chapter outlines the development procedures for the motor's DT model using a model-based design approach. It covers the modeling techniques, tools, and methodologies employed to represent the motor's behavior and performance accurately.

- Chapter 5: Powertrain Digital Twin Modeling Framework

This chapter presents the final hybrid model, which integrates both the motor model-based model and the drive system data-driven model. It discusses the challenges and solutions associated with combining these components into a cohesive DT, ensuring accurate interaction and data exchange.

- Chapter 6: Conclusions and Future Work

The final chapter summarizes the research findings, discusses the implications of the study, and outlines potential areas for future research.

2 Literature Review

2.1 Background

Historically, the development of automotive and aerospace systems relied on empirical engineering methods. However, the rising demands for higher performance, operational "self-awareness," and independence from external support have necessitated the adoption of innovative engineering procedures[18]. The advent of DT has introduced novel testing and development environments to meet these new demands.

A DT for EV is a virtual model that accurately replicates the physical attributes, systems, and real-time operational data. This digital counterpart integrates information from sensors, onboard diagnostics, and external data sources to mirror the vehicle's performance and condition [19]. By simulating and analyzing this data, manufacturers, operators, and service providers can monitor the EV's status, predict maintenance needs, optimize performance, and enhance decision-making throughout the vehicle's lifecycle. Despite the numerous advantages that DT technology brings to the EV industry; it is still in the early stages of mastering this application [20].

From a manufacturer's perspective, several challenges slow DT adoption. First, implementing DTs demands high upfront investments in technology, software, and skilled personnel [21]. Second, accurately modeling complex EV systems and ensuring smooth integration between physical and digital components requires advanced expertise and resources [22]. Third, concerns over data security and intellectual property protection make manufacturers cautious, as extensive data exchanges risk exposing sensitive information to cyber threats [23]. Finally, the absence of standardized protocols and industry-wide frameworks adds uncertainty, deterring commitment to technologies that may quickly evolve or become obsolete. Collectively, these factors lead to a gradual embrace of DT technology in EV manufacturing [24].

The challenge of modeling intricate EV systems arises from the complex interplay of electrical, mechanical, and software components. This complexity, along with the lack of standardized frameworks, has resulted in fragmented approaches and inconsistent outcomes in DT development, limiting scalability, interoperability, and adoption. This chapter overviews DT modeling frameworks, methodologies, tools, and applications – with a focus on techniques, software platforms, and DTs designed for EV propulsion drive systems – to provide a solid foundation for advancing DT technology in the field.

2.2 Digital Twins Modelling Techniques

The core of DT technology lies in its ability to create detailed, dynamic models that mirror real-world objects or processes, providing unparalleled insights and control over complex systems [25]. Modeling serves as the fundamental or initial step in developing a DT, as it establishes a comprehensive digital representation of the physical system. This model forms the foundation upon which all subsequent analyses, simulations, and optimizations are built, ensuring that the DT can accurately reflect and respond to the behaviors and conditions of its real-world counterpart. According to DT modeling techniques, they can be divided into three primary categories, as detailed in the following subsections.

2.2.1 Model-Based Digital Twin

The model-based simulation (MBS) approach is a structured methodology used for establishing requirements, designing, analyzing, and validating complex systems. Central to MBS is the use of models in system design. Physical systems, whether found in nature, test environments, or applications, consist of interconnected components that perform tasks or various functions. Using MBS to simulate a physical system involves analyzing the system's mechanisms through fundamental physical laws and engineering principles. The effectiveness of MBS is rooted in a profound understanding of the system or process and can benefit from established scientific relationships. Model-based DT extends MBS by incorporating enhanced sensory data and AI tools [26].

Based on literature, various examples of model-based DTs and the platforms employed for their development across different applications. For instance, Madni et al. [27] implemented DT technology in a vehicle model-based system using the planar mechanics open-source library. Bachelor et al. [28] conducted a case study on a model-based DT for an ice protection system in a regional aircraft using the Dassault Systems Dymola platform. Magnanini and Tullio [29] proposed an analytical model-based DT for evaluating the performance of a railway axle manufacturing system based on Markovian System representation. Zheng and Sivabalan [30] developed a DT for a cyber-physical system (CPS) of a 3D printer using a Windows Presentation Foundation (WPF) Application and .Net framework 4.5 in Visual Studio, based on a tri-model approach for product-level development. Ward et al. created a model-based DT system for large-scale CNC machine tools using the MATLAB/Simulink platform. Yang et al. [31]developed a model-based DT for an aero-engine disk to detect unbalanced and crack failure online, utilizing the ANSYS simulation platform. Woitsch et al. [32] proposed a meta-model for a model-based DT environment to bridge the gap between manufacturing and product usage, using the ADOxx meta-modeling platform.

These examples show that creating a model-based DT of a system is closely linked to physical systems that can be modeled and generally relies on conventional modeling and simulation platforms alongside AI techniques and IoT tools. Although model-based DTs are widely used across various applications, they face certain limitations, especially with highly complex systems. The primary disadvantage of model-based DTs is their inability to manage infinite complexity, often requiring simplification.

2.2.2 Data-Driven Digital Twins

Implementing DTs allows operators to monitor production, test deviations in a virtual environment, and enhance the security of process industries. As process data significantly increases, traditional model-based methods are insufficient for describing the state space of complex systems. Thus, data-driven modeling technology has emerged as a viable solution for modeling DTs. Data-driven modeling involves analyzing system data to identify connections between variables (input, internal, and output) without explicit knowledge of the system's physical behavior. Compared to traditional empirical models, these methods represent a significant advancement across numerous applications [33]. Data-driven modeling relies on substantial and sufficient data to describe the underlying system, enabling tasks such as classification, pattern recognition, associative analysis, and predictive analytics.

Studies indicate that data-driven DTs are widely applied in many areas, especially in complex systems. Wang et al. [34] developed a data-driven DT framework for a three-domain mobility system involving humans, vehicles, and traffic using the Amazon

Web Services (AWS) platform. Gao et al. [35] employed the MATLAB/Simulink platform to construct an anomaly detection framework for monitoring abnormal behaviors in a data-driven DT-based cyber-physical system. Coraddu et al. [36]created a data-driven DT for ships to estimate speed loss and marine fouling using numerous onboard sensors and the IBM Engineering Lifecycle Management (IBM-ELM) platform. Mykoniatis and Harris [37] developed a data-driven DT for an automated mechatronic modular production system, focusing on condition monitoring, design decisions, testing, and validating physical system behavior, using the AnyLogic Simulation platform. Blume et al. [38] designed a data-driven DT for cooling towers to improve system understanding and performance prediction, employing tools like KNIME and Microsoft Excel. Kim et al. [39] developed a data-driven DT for on-load tap changers (OLTC) to monitor health and detect faults using a numerical algorithm for subspace state-space system identification (N4SID). Major et al. created a Java-based data-driven 3D graphical DT platform for smart city applications, backed by a real case study in Norway.

2.2.3 Hybrid Driven Digital Twins

Hybrid DT models integrate the strengths of both model-based and data-driven approaches to address their limitations. By combining physical models with data-driven techniques, hybrid modeling produces a more robust and comprehensive DT [40]. This approach is particularly advantageous for complex systems where purely model-based or data-driven methods are inadequate. Hybrid models utilize detailed physical insights from model-based methods and the pattern recognition capabilities of data-driven techniques, resulting in a more accurate and adaptable simulation. A significant advantage of hybrid DT models is their ability to leverage the detailed physical insights from model-based methods while incorporating the pattern recognition capabilities of data-driven techniques [41].

This combination allows for a more accurate and adaptable simulation of complex systems. For instance, in the context of naval shipboard power and energy systems, hybrid modeling techniques combine analytical and data-driven approaches to offer the benefits of both types of models [42]. The integration of reduced-order models with Bayesian state estimation has been demonstrated to enable the creation of data-driven physics-based DTs, which can rapidly adapt and quantify uncertainties for large complex systems such as unmanned aerial vehicles [43]. This approach ensures that the DT can accurately reflect the physical system's state and predict its future behavior under various conditions. The dynamic runtime integration of new models in DTs allows for the replacement of individual models without disrupting the operation of cyber-physical systems, enabling seamless updates to the overall model structure [44].

2.3 Digital Twin Software and Platforms

DT technology necessitates advanced platforms capable of generating digital simulations of physical entities. This software monitors asset performance and runs simulations to predict potential outcomes or maintenance needs. Various software platforms support DT creation, as summarized in the review [paper II].

- *AWS IoT.* Amazon’s platform which is primarily used for remote monitoring, facilitating data exchange between a remote emulation or simulation and the physical twin. AWS combines AI and IoT to make devices more intelligent, allowing models created in the cloud to be deployed to devices where they run twice as fast compared to other offerings.

- *Siemens Sim Center.* Siemens offers a suite of tools under the Sim center portfolio, including 3D CAE, system simulation, and test solutions. It is designed to handle complex engineering challenges and is widely used in the automotive industry for building DT of EV systems.
- *Ansys Twin Builder.* This platform allows engineers to create simulation-based DTs – digital representations of assets with real-world sensor inputs. It is primarily used in industrial applications for design, testing, predictive maintenance, and optimization. Ansys Twin Builder features various sub-platforms tailored for EV applications.
- *PTC ThingWorx.* This industrial innovation platform offers powerful tools for building DTs, including connectivity, analytics, and user interface creation. It is used extensively in the manufacturing and automotive sectors.
- *Dassault Systèmes CATIA.* CATIA offers a comprehensive engineering solution that supports the creation of DTs. It is used to model and simulate the behavior of complex systems, including those in the automotive industry.
- *Matlab/Simulink.* Simulink, an add-on product to MATLAB, provides a graphical environment for simulation and Model-Based Design of multidomain dynamic and embedded systems. It is widely used in the automotive industry to develop DTs of EV systems, allowing engineers to model, simulate, and analyze the performance of EV components.
- *Altair HyperWorks.* This simulation-driven design platform offers a suite of tools for finite element analysis (FEA), computational fluid dynamics (CFD), and multi-body dynamics. It is used to create DTs to optimize the performance and reliability of EV systems.
- *Bentley Systems iTwin.* This platform provides infrastructure DTs that integrate engineering, reality, and IoT data. It is used to improve the performance and resilience of various systems, including transportation and EV infrastructure.
- *Hexagon's HxGN.* Hexagon offers solutions for creating DTs that support various industries, including automotive. Their software helps optimize EV systems' design, operation, and maintenance.
- *NVIDIA Omniverse.* It is a cloud-native platform designed for real-time, AI-powered simulation and collaboration, particularly in the development of DT. It integrates with industry-standard tools and supports physics-based rendering, making it a robust solution for high-fidelity DT modeling. The platform is widely adopted in automotive applications, facilitating seamless data fusion from multiple sources, enabling interactive scenario testing, and providing realistic 3D visualization of the vehicle systems.

While various platforms offer distinct capabilities for developing DTs, MATLAB/Simulink stand out as the most versatile and suitable options for research and broader applications. Their flexibility and comprehensive toolsets enable modeling and simulation across a wide range of industries. In aerospace, for instance, it is used to model and simulate aircraft dynamics and control systems. In industrial automation, these tools optimize manufacturing processes and equipment maintenance. For energy systems, it is instrumental in modeling power grids and renewable energy sources. In the biomedical field, researchers simulate physiological responses for medical devices, enhancing patient outcomes. In addition to that it has high flexibility in supporting diverse modeling approaches – whether model-based, data-driven, or hybrid – ensures accurate replication of complex systems. One of MATLAB/Simulink's key advantages is its real-time simulation capability, a critical feature for DTs to mirror and interact with physical systems in real-time, making it superior to conventional simulation platforms. This real-time performance enables effective propulsion system monitoring, control, and optimization.

Furthermore, MATLAB/Simulink's open architecture and general-purpose design make it applicable to a wide range of fields, offering more versatility than other platforms that are often tailored for specific applications and operate as closed systems limited to their manufacturers. It also supports extensive communication with other platforms and offers comprehensive data exchange protocols, enhancing interoperability. With its strong scientific support, extensive libraries, powerful AI tools, and widespread availability, MATLAB/Simulink represents a superior choice for developing adaptable, high-performance DTs.

2.4 Overview of Digital Twins for EV Propulsion Drive Systems

The propulsion drive system is the fundamental component of the EV, requiring efficiency, reliability, and cost-effectiveness to ensure optimal performance. As depicted in Figure 5, the system comprises two fundamental subsystems: the electrical and the mechanical. These components work together seamlessly to convert electrical energy into mechanical motion, showcasing the efficiency and sophistication of modern EV propulsion. The electrical subsystem includes key components responsible for managing and manipulating electrical energy. At its core is the electric motor, which transforms electrical energy from the vehicle's Energy storage system (ESS) into rotational mechanical energy. Advanced power electronics, such as inverters and controllers, regulate the flow of electricity to the motor, optimizing performance, efficiency, and responsiveness [45]. The mechanical subsystem transfers the mechanical energy produced by the electric motor to the wheels, comprising the differential unit, which divides power between the wheels, and the gearbox system, which adjusts the motor's output to changing driving conditions. These mechanical elements ensure effective torque delivery, providing a dynamic and controlled driving experience. Figure 6 illustrates the EV propulsion system.

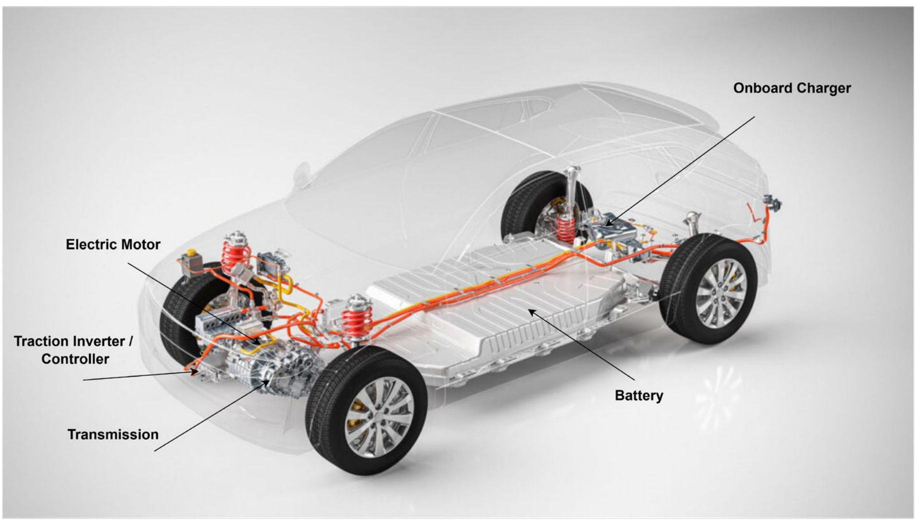


Figure 5. EV propulsion drive system.

DTs for EV drive systems are commonly employed for health monitoring, diagnostics, prognostics, optimization, scenario analysis, and lifetime prediction. They can be developed at various levels, including the system, subsystem, individual components, and other assets.

2.4.1 Battery Energy Storage System (BESS)

Many researchers consider the battery an integral component of the EV drive system, including its associated management system. The maximum driving distance of an EV is primarily determined by the battery capacity – the greater the capacity, the longer the driving range. In 2019, Lithium-ion batteries (LIBs), including variants such as Lithium Nickel Manganese Cobalt (NMC) oxide and Lithium Nickel Cobalt Aluminium (NCA) oxide, dominated the EV battery market with nearly 96% market share. The performance of the EV battery directly influences the design and operation of the traction inverter, which is critical for managing the flow of electrical energy to the motor [46].

Advanced battery technologies are continually being developed to improve capacity, efficiency, and safety. Effective thermal management systems are crucial to prevent overheating and ensure optimal performance. These systems help maintain the battery within an ideal temperature range, enhancing its longevity and reliability.

According to the review [Paper I], DT can significantly enhance battery technology in several key areas. They allow for precise simulation of battery performance, improving design and thermal management. Real-time battery health monitoring enables predictive maintenance, extending battery lifespan and reliability. Integration with advanced battery management systems optimizes energy efficiency and safety [47]. Additionally, DTs accelerate innovation by allowing virtual testing of new materials and configurations, reducing development time and costs.

2.4.2 Electric Motor

The electric motor is a critical component of an EV, responsible for converting electrical energy into mechanical energy and vice versa during regenerative braking operation. Each EV motor is tailored to meet specific vehicle requirements, characterized by the ability to maintain consistent power across various speeds, generate high torque at low speeds for hill climbing, exhibit high torque and power density, have a compact design, and, most importantly, deliver high operational efficiency [48]. Designing and manufacturing EV motors involves numerous challenges due to different environmental conditions, such as extreme ambient temperatures (-30 to 60 °C), shocks, and strict constraints on size, weight, safety, and reliability[49]. Currently, several types of electric motors are used in EV applications, including Permanent Magnet Synchronous Motors (PMSM), Brushless Direct Current Motors (BLDC), Induction Motors (IM), Switched Reluctance Motors (SRM), Hybrid Excitation Synchronous Motor (HESM), and Permanent Magnet Assisted Synchronous Reluctance Motor (PMSynRM) among others. Each motor type offers unique characteristics and advantages, catering to the diverse needs of different EV applications [50].

DT technology can significantly enhance the development and application of EV motors. By creating a virtual replica of the motor, DTs can simulate performance under various conditions, helping to optimize design and control strategies. This technology enables real-time monitoring and predictive maintenance, reducing the risk of unexpected failures and extending the motor’s lifespan [51]. Additionally, DTs can improve efficiency by dynamically adjusting operational parameters based on real-time data, ensuring optimal performance [52]. Many research studies have adopted DT for electric motors. Rjabtšikov et al. [53] proposed a DT model for fault detection in an AC 3-phase induction motor (IM). They implemented inter-turn short circuit fault detection within the motor's DT. The emulator was built using historical data and a mathematical motor model, and it utilized Unity 3D combined with ROS service for online condition monitoring. In this study, the DT model served as a virtual sensor for the physical motor model. Toso et al. [54] applied a DT to optimize the performance of an EV motor, specifically focusing on estimating driving torque and cooling control. They first conducted thermal and electromagnetic finite element analysis (FEA) of an EV induction motor to gather necessary data for optimization. The DT model of the motor was constructed using a micro lab box as a system-level DT. Katukula et al. [55] developed a DT system to monitor and analyze the conditions of an IM. This DT system measured the motor's current using sensors, with the collected data fed into a simulation finite element model (FEM). This approach provided a better understanding of the motor's thermomagnetic behavior and allowed for predicting potential faults. Rassõlkin et al. [56] outlined a methodology for collecting data from an AC motor drive system based on an empirical performance model. This data was used to develop a DT model. Unity 3D was used as the host environment to simulate and visualize the motor's DT model, which estimated the drive system's performance. Brandtstaedter et al. [57] presented a DT model for fault detection in an electric drivetrain's 50 MW permanent magnet synchronous motor (PMSM). They numerically simulated the motor and developed a framework for fault identification. Their model tested and verified unbalance detection and temperature prediction in the rotor system. Jitong et al. [58] developed a DT model of a 3-phase IM to facilitate the motor design and monitor the motor's regular operation, aiming to cover its entire lifecycle and reduce maintenance time. A 3D simulation model of the physical motor was built using 3ds Max, with the DT constructed based on the simulation model using Unity 3D software. Data acquisition between the real and digital models was managed using an SQL server. Venkatesan et al. [59] introduced a DT system of an EV-PMSM drive system for health monitoring and prognosis. They monitored outputs such as casing temperature, winding temperature, time to refill bearing lubricant, and percentage deterioration of magnetic flux to calculate the remaining useful life (RUL) of the permanent magnet (PM). Two approaches were presented for motor health monitoring: one for in-house monitoring and the other for remote monitoring. The DT model was built using MATLAB/Simulink, incorporating Artificial Neural Networks (ANN) and fuzzy logic to map system inputs.

2.4.3 Inverter / Motor Control Unit (MCU)

The intricate interaction between inverter technology and control strategies is at the core of electric propulsion systems. The inverter, essential in EVs and HEVs, converts DC power from the battery into AC for motor operation and reverses the process during regenerative braking. Its effectiveness depends on the seamless integration of hardware and software, with the hardware enabling power conversion and control strategies managing the flow of electrical energy to optimize motor performance. Insulated Gate Bipolar Transistors (IGBTs) have been fundamental in early inverter designs and are valued for their reliability and cost-efficiency in DC/AC conversion. However, their efficiency diminishes in high-power applications due to increased switching losses, limiting their effectiveness at higher power levels despite being suitable for a wide range of automotive uses [50].

The introduction of Silicon Carbide (SiC) and Gallium Nitride (GaN) semiconductor materials has significantly advanced inverter technology. SiC inverters, known for their high thermal conductivity and electron mobility, offer improved efficiency, particularly in high-temperature conditions. This makes SiC-based inverters ideal for demanding automotive applications, supporting high power ranges with superior efficiency and thermal management. Studies have shown that upgrading from a conventional 400V IGBT inverter setup to an 800V DC bus with SiC devices can reduce an EV's energy consumption by about 5%, effectively increasing its driving range. GaN-based inverters excel in achieving higher switching frequencies, which helps reduce the size and weight of inverter systems – an important factor in EV design. Although GaN inverters are typically used in lower power ranges, up to tens of kW, their efficiency and compactness make them an attractive option for lightweight or compact EVs [60].

The advancement of DT technology for EV inverters has had a significant impact. Health monitoring, fault diagnosis, performance optimization, and lifetime estimation of semiconductors are the main prospective functions of DTs for inverters, as the literature highlights. Milton et al. [61] proposed a DT of a power converter running on a field programmable gate array (FPGA) for online diagnostic analysis. Wunderlich and Santi [62] developed a data-driven DT model of a power electronic converter based on a dynamic neural network for condition monitoring. Liu et al. [63] proposed a model-based DT of a power electronic converter for condition monitoring. Wu et al. [64] proposed a DT approach for a single-phase inverter to identify degradation parameters. Shi et al. [65] proposed a DT method for IGBT parameter identification of a three-phase DC/AC inverter based on a particle swarm optimization algorithm for circuit condition monitoring. Liu et al. [66] developed and experimentally validated a DT of an automotive traction drive system, combining a FEM-based PMSM model with a SiC-inverter circuit simulation.

2.4.4 Mechanical Transmission

The transmission system in an EV transfers mechanical power from the motor to the wheels. Although the mechanical power source (the electric motor) and the wheels could theoretically be connected directly, their RPMs do not align well. The gear trains in the transmission system play a crucial role in allowing them to rotate at different speeds. The difference in RPMs between the motor and the wheels is managed through gear ratios, which directly impact vehicle performance, including acceleration, top speed, and energy consumption. Higher gear ratios produce greater acceleration but lower top speeds, while optimal gear ratios help improve battery life and driving range [67].

Initially, research in EVs focused on improving electric drives, storage technologies, and charging infrastructure. However, the role of transmission systems has since been recognized as crucial for enhancing drivetrain efficiency and overall vehicle performance. Unlike conventional ICE vehicles, EVs do not require complex engine setups, clutches, or manual gear shifts. This simplicity and fewer parts lead to increased transmission efficiency in EVs [68].

Despite the simplicity of the EV transmission system, it can still experience faults that require thorough analysis. Such faults can lead to a series of problems in the entire EV propulsion system. This is where the role of DT technology becomes critical. They provide several key benefits for managing and optimizing EV transmission systems. Firstly, DTs enable continuous monitoring of the transmission system's performance, allowing for early detection of anomalies and potential faults, which can prevent significant issues through timely maintenance. Secondly, by utilizing historical data and predictive algorithms, DTs can forecast when transmission components are likely to fail, facilitating proactive maintenance and reducing downtime [69].

2.5 Current Challenges and Research Gap

Based on the review analysis, the main research gap can be identified as the lack of standardized frameworks and methodologies in developing DTs for the automotive industry. This gap presents a significant challenge for creating consistent and scalable DT solutions. Without standardized approaches, there is variability in how DTs are implemented, which hinders the broader adoption and integration of this technology across different automotive applications. Establishing best practices and a unified framework for DT development is essential to ensure consistent performance, interoperability, and reliability. This would streamline the development process and enable seamless communication between digital and physical systems, thereby promoting a more widespread and efficient implementation of DT technology in the automotive industry.

The second critical issue is real-time data integration and analysis, which remains underutilized despite being a key feature of DTs. The ability to efficiently process large volumes of data, ensure accuracy and consistency, and develop algorithms capable of rapid decision-making and predictive analytics is still a gap that needs to be addressed. Moreover, as DTs become more widespread, the challenge of interoperability and integration with legacy systems arises. Ensuring that DTs can seamlessly interact with existing technologies requires the development of universal standards and protocols for data exchange.

The lack of research in this area has made ensuring compatibility across different platforms challenging, making integration a pressing issue. Balancing accuracy and computational time present a third significant challenge in DT modeling. While high-fidelity models provide detailed and precise simulations, they often demand substantial computational resources, resulting in longer processing times. This becomes problematic for real-time or near-real-time applications. Research into advanced modeling techniques – such as reduced-order modeling, surrogate modeling, and machine learning-based approximations – is needed to address this. The mentioned techniques aim to simplify complex models while preserving essential dynamics, reducing computational demands without significantly sacrificing accuracy. Developing and validating these methods is crucial for making DTs both practical and efficient, particularly in time-sensitive automotive applications.

3 Physical Propulsion Drive System Overview

The physical system serves as the foundation of any DT system, acting as the real-world counterpart that the DT replicates. It consists of the tangible components, subsystems, or processes that are being modeled. In the context of a DT, the physical system is equipped with sensors and data acquisition technologies to continuously capture real-time data on its performance, condition, and environmental factors. This data forms the basis for the digital model, allowing the DT to accurately simulate the behavior of the physical system under various conditions.

3.1 EV Propulsion Drive System

The propulsion system under study was taken from ISEAUTO (the first Estonian self-driving vehicle), which was developed based on the Mitsubishi i-MiEV 2010 propulsion system, incorporating Y4F1 PMSM [20]. It was designed as a last-mile vehicle, operating at limited speeds up to 20 km/h, with the transmission system ensuring the motor functions within its most efficient range. The concept of ISEAUTO was to give a second life to a legacy motor drive system in a new EV application, reducing waste and promoting sustainable practices.

Converting ISEAUTO's physical propulsion drive system into a test bench in a laboratory setting is crucial to facilitating testing and development. This conversion allows for controlled experimentation and analysis, enabling the detailed study of system components and their interactions under various conditions.

The developed test bench comprises several key components replicating the vehicle's powertrain in a controlled lab environment. These components include the PMSM motor, inverter/controller, and transmission system, all mounted on the test rig. The setup has loading emulators that simulate real-world driving conditions and advanced sensors and data acquisition systems to monitor and record performance metrics such as speed, current, and voltage. A controlled battery emulator system powers the setup. The testbench is controlled through a PC interface module, including different platforms of the setup components. The following Figure 6 shows the propulsion drive system test bench.

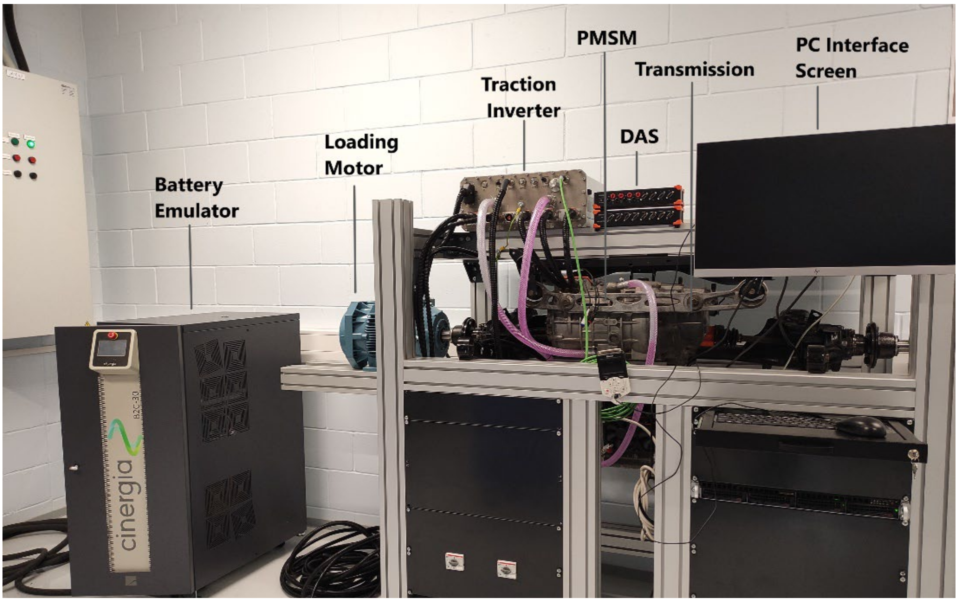


Figure 6. EV propulsion drive system testbench [70].

3.1.1 Battery Emulator

CINERGIA B2C+ battery emulator is a highly advanced unit designed to simulate the behavior of batteries, making it an essential tool for testing and developing energy storage applications. It features a regenerative AC/DC converter, allowing efficient energy conversion with minimal losses. It has a wide output range (20–750 VDC), making it adaptable to various battery configurations. The unit's power and current capabilities are highly flexible, as its outputs can be configured in series or parallel to meet different testing requirements. The system is controlled through a PC interface, with sophisticated software that provides full control over the emulator's parameters, ensuring precision in replicating battery characteristics. This setup allows for detailed testing and optimization of power systems. Additionally, the device supports communication through multiple industry-standard protocols, including CAN bus, Modbus, and Ethernet Open, enabling seamless integration with other systems and equipment.

3.1.2 Traction Inverter / Controller

The motor drive system is an ABB HES880 model, with a 55 kW heavy-duty, water-cooled electric drive. This inverter is controlled via a PC interface and supports data exchange through CAN bus or Ethernet. It operates in both inverter and generating modes, employing a vector control strategy.

3.1.3 Electric Motor

The electric traction motor used in the ISEAUTO is a 25 kW to 47 kW peak Mitsubishi Y4F1 water-cooled interior PMSM designed for high efficiency and performance in demanding environments such as EVs. The water-cooling system ensures optimal thermal management, allowing the motor to operate consistently under high loads while maintaining stable temperatures, enhancing its durability and efficiency. This motor has an integrated resolver unit mounted on its internal shaft, providing precise speed and position measurement.

3.1.4 Transmission

The F1E1A is a four-gear, double-stage transmission unit designed for efficient power transmission. The first gear consists of a pinion gear mounted on the input shaft, which is directly coupled to the rotor of the vehicle's traction motor, ensuring seamless power transfer from the motor to the transmission system. The second and third pinion gears are mounted on a shared intermediate shaft, allowing smooth transitions between gears. The fourth gear is directly connected to both the output shaft and the differential unit, facilitating power distribution to the vehicle's wheels and enabling effective propulsion.

3.1.5 Load Emulators

Two 7.5 kW IMs are installed on either side of the differential unit to simulate the movement of the vehicle's wheels. These motors are powered by two ABB frequency converters, which precisely control their operation. The frequency converters enable synchronization with the vehicle's dynamics, allowing the motors to accurately replicate the behavior of the wheels under various driving conditions.

3.1.6 Data Acquisition System

A data acquisition system (DAS) is an advanced tool for comprehensive electrical and mechanical power analysis, essential for testing and developing different EV systems. The combined DEWESOFT SIRIUS XHS and DEWE-43A data acquisition module used in this study offers a highly versatile and performance-driven solution for capturing key measurement signals, including voltage, current, speed, torque, power, and other critical parameters.

The system comprises two complementary components: the SIRIUS XHS, which is primarily dedicated to high-speed voltage and current measurements, and the DEWE-43A, which focuses on capturing torque and speed data. The SIRIUS XHS features 8 high-speed analog input channels, supporting sampling rates of up to 15 MS/s per channel, making it ideal for high-resolution, high-frequency electrical data. It accommodates a wide input range of up to ±1000 V, allowing for accurate current and voltage measurements during EV testing. In parallel, the DEWE-43A provides 8 additional analog channels with a sampling rate of 200 kS/s per channel, optimized for mechanical parameter acquisition such as torque and speed, where a ±10 V input range is sufficient for most sensor outputs.

Both components have 24-bit Sigma-Delta ADC converters, ensuring precision across the measurement spectrum. The system supports galvanic isolation on each input channel to protect against interference and ensure data integrity. Advanced synchronization options, including GPS, IRIG, NTP, and PTP (IEEE 1588), are available for the SIRIUS XHS, while the DEWE-43A offers internal synchronization via USB, ensuring seamless integration between the two modules. Data interfaces, including USB 3.0 and CAN, allow for robust communication across different platforms.

4 EV-PMSM Digital Twin Modelling Framework

This chapter details the DT modelling framework for the EV PMSM utilized in this study, providing a comprehensive overview of the methodologies and findings presented in research papers [IV], [V], and [VI].

The core issue lies in the motor being part of a pre-existing system with constrained data availability, making accurate modelling a more demanding task requiring deeper investigation and advanced techniques. The modelling process involves a structured approach encompassing several critical steps to ensure the creation of a highly accurate and functional virtual model of the PMSM that is capable of real-time interaction with its physical counterpart. The process begins with system identification, where the key characteristics and dynamic relationships of the PMSM are defined despite the limited known parameters. This is followed by parameter estimation, where unknown variables are determined through experimental testing and system identification methods.

Once the parameters are established, a suitable modeling technique is selected to balance accuracy with computational efficiency, addressing the complexities associated with modeling and real-time performance. The digital model then undergoes rigorous validation, testing its performance against the real motor under various operational conditions. Once validated, the model is deployed to a DT framework using a real-time target machine, demonstrating its capability to mirror the physical motor in real time and provide valuable insights for continuous condition monitoring.

4.1 PMSM Model Identification

Identifying the PMSM model is a critical step in developing an accurate virtual model that reflects its actual behavior. This process involves identifying known parameters declared by the manufacturer as well as estimating unknown parameters. The manufacturer's known motor parameters are listed in Table 4 [71], [72].

Table 2. Y4f1 PMSM nameplate.

Parameter	Description	Value	Unit
p	Number of pole pairs	4	-
N_r	Rated speed	3000	rpm
P_r	Rated output power	25	kW
P_{Max}	Maximum output power	47	kW
T_{Max}	Maximum torque	180	N·m

The known motor parameters, listed in Table 4, include key ratings that offer a foundational understanding of the PMSM's performance; however, they are insufficient for developing a high-fidelity digital model. Critical lumped parameters and internal characteristics, which directly govern the motor's dynamic behavior, such as resistance, inductance, and magnetic flux linkage, remain unclear. These unknown parameters require further investigation and estimation technique.

4.2 Unknown Parameters Estimation

The conventional approach for measuring PMSM lumped parameters typically involves open-circuit voltage and short-circuit tests [4]. However, these tests pose significant risks to the motor windings and the traction drive for high-power machines such as EV motors. Various alternative methods have been developed for PMSM parameter estimation to address these challenges. These various methods can be broadly classified into several categories based on their approach and application. Steady-state and dynamic measurement techniques involve analyzing the motor's behavior under specific operating conditions, such as constant velocity or time characteristics, providing a safer alternative for high-power motors but they may lack precision in capturing complex motor dynamics. Advanced computational methods include techniques like the Extended Kalman Filter (EKF) and optimization algorithms (e.g., Box's method, simulated annealing particle swarm optimization) that offer high precision and the ability to handle complex, real-time parameter estimation. However, these methods require significant computational resources and expertise in algorithm development.

The proposed method for estimating PMSM parameters combines system identification theory with an optimization search algorithm achieving a balance between high precision and computational resources. It involves identifying the unknown parameters of the motor's mathematical model using actual experimental values, such as stator voltages, currents, and rotational speed. The general PMSM mathematical model is employed, combined with an optimization search algorithm to determine the optimal values for the unknown parameters. The algorithm iteratively adjusts these parameters to match the experimental data, ensuring that the model accurately reflects the motor's real-world behavior.

4.2.1 Stator Resistance

PMSM stator resistance is measured through a DC test, in which the motor's stator windings were energized with a low DC voltage. Since the applied voltage is DC, no back electromotive force (EMF) is generated because the rotor remains stationary. This eliminates the inductive effects and allows the resistance to be calculated purely based on the relationship between the applied DC voltage and the resulting current through the stator windings, according to Ohm's law.

4.2.2 Inductances and Permanent Magnet Flux

While the stator resistance could be measured directly, determining the inductance and linkage flux requires a more sophisticated approach. The proposed parameter estimation method is based on matching the experimental data from the motor test bench with the generalized d-q mathematical model of PMSM. Consequently, no-load tests were conducted at different operating speeds within the motor's rated speed range. Each test's recorded data was systematically archived in MATLAB data files to ensure its seamless integration into the estimation algorithm.

The estimation algorithm was developed using a MATLAB script that implements the mathematical representation of the PMSM as objective functions. The key unknown parameters in the motor model are the stator d-q inductances and the permanent magnet flux linkage. The goal of the algorithm is to estimate these parameters by fitting the PMSM equations to the experimental data collected. The MATLAB script processes the measured voltage, current, and speed data obtained from the physical motor.

The input voltage and stator currents are transformed into the d-q rotating reference frame using park transformation, which aligns with the mathematical model's framework and facilitates accurate calculations. The rotor speed and position are also incorporated into the algorithm using data from the resolver, ensuring that all relevant dynamic aspects of the motor are considered. To enhance the accuracy of parameter estimation, Ant Lion Optimization (ALO) algorithm was integrated into the MATLAB script. ALO algorithm, developed by Seyedali Mirjalili [73] is a meta-heuristic optimization technique that simulates the hunting mechanism of ant lions in nature. ALO algorithm searches for the optimal values of the unknown motor parameters by minimizing the difference between the calculated d-q currents from the PMSM equations and those measured experimentally. The algorithm begins with initializing a random population of ants, which move around ant lions in the search space. The fitness of each ant's position is then evaluated relative to the ant lions. Based on these evaluations, each ant's position is updated using a random walk, guided by a roulette wheel selection, until the best ant lion, referred to as the elite, is identified. The position of the best new ant lion is subsequently evaluated and updated according to their objective values. The algorithm identifies the current best ant lion position if the stopping criterion is met. If not, the process loops back to the fitness evaluation phase, using the best ant lion position obtained so far, and continues until the maximum number of iterations is reached.

The optimization was guided by the following objective functions and constraints:

A. objective function:

$$[\,\mathrm{F} = \min_{L_d, L_q, \psi_{pm}} \left\{ \sum_{\mathrm{k}=1}^{\mathrm{N}} \left[\, I_{d,\mathrm{meas}}(k) - I_{d,\mathrm{sim}}\left(k;\, L_d,\, L_q,\, \psi_{pm}\right) \right]^2 + \left[I_{q,\mathrm{meas}}(k) - I_{q,\mathrm{sim}}\left(k; L_d, L_q, \psi_{pm}\right)\right]^2 \right\}] \tag{1}$$

where $(I_{d,\mathrm{meas}}(k))$ and $(I_{q,\mathrm{meas}}(k))$ represent the measured direct and quadrature axis currents at the (k)- th sampling instance, while $(I_{d,\mathrm{sim}}(k; L_d, L_q, \psi_{pm}))$ and $(I_{q,\mathrm{sim}}(k; L_d, L_q, \psi_{pm}))$ denote the corresponding emulated currents based on the lumped parameters.

The objective is to minimize the sum of squared errors between the measured and simulated currents over *N* sampling instances.

B. optimization equalities:

$$\begin{cases} I_{d,\mathrm{meas}} = I_{d,sim} \\ I_{q,\mathrm{meas}} = I_{q,sim} \end{cases}; \tag{2}$$

$$\begin{cases} V_{d,\mathrm{meas}} = V_{d,sim} \\ V_{q,\mathrm{meas}} = V_{q,sim} \end{cases}; \tag{3}$$

$$\omega_{e,meas} = \omega_{e,sim} \tag{4}$$

C. constraints:

$$
\begin{gathered}
0 < L_q < 0.1 \\
L_d < L_q \\
0 < \psi_{pm} < 1
\end{gathered}
\tag{5}
$$

4.2.3 Estimation Results

The ALO algorithm provided slight variations in the parameter values depending on the different test conditions. Table 3 presents the detailed parameter data for each test as determined by the ALO algorithm, along with the average value for each parameter obtained across all tests based on the updated data from paper [III].

Table 3. Parameter estimation results.

Test	Speed (rpm)	Ld (mH)	Lq (mH)	ψ_{pm} (Wb)
A	300	2.01936	2.58108	0.09427
B	500	1.99263	2.54433	0.09302
C	700	1.83373	2.47209	0.08913
D	900	1.78835	2.32286	0.08819
E	1000	1.77220	2.25232	0.08450
Average		1.90000	2.40000	0.08700

The observed fluctuations in the d-q inductance values indicate a decrease in the motor's inductance as speed increases. This reduction can be attributed to magnetic saturation within the motor's core, where higher flux densities at elevated speeds cause the core material to reach its saturation point more quickly. Additionally, the frequency-dependent nature of inductance plays a role, as increased operating frequencies at higher speeds exacerbate the skin and proximity effects in the stator windings, further diminishing the inductance. The permanent magnet flux values also exhibit a slight reduction with increasing speed that could be attributed to the effects of the flux weakening control at higher speeds. However, it is important to note that these variations are still within the acceptable level of 5%. The overall motor parameters are illustrated in the following Table 4.

Table 4. PMSM motor parameters.

Parameter	Description	Value	Unit
R_s	Stator resistance	0.010087	Ω
L_d	d-axis inductance	1.90000	mH
L_q	q-axis inductance	2.40000	mH
ψ_{pm}	PM linkage flux	0.08700	Wb

4.3 EV-PMSM Digital Modelling

The next step in motor DT development is constructing an accurate and efficient digital model replicating the motor's behavior under various operating conditions. Given that the motor parameters have been accurately estimated, using a physics-based modeling approach is the most appropriate method. Under the category of model-based modeling, there are several modeling techniques available, each with its strengths and trade-offs. Among the most common methods are Finite Element Analysis (FEA), d-q Equivalent Circuit (d-q EC), and Magnetic Equivalent Circuit (MEC).

- *FEA* offers the highest level of accuracy by solving the motor's electromagnetic field equations in great detail. This method is particularly effective for capturing complex phenomena such as magnetic saturation and non-linear material properties. However, FEA is computationally intensive, making it less suitable for real-time applications where quick response times are essential.
- *d-q EC* is widely used in EV applications due to its balance between accuracy and computational efficiency. This method simplifies the analysis of the motor's dynamic behavior by transforming the motor's three-phase currents and voltages into a two-axis (d-q) reference frame. The d-q model is particularly advantageous for real-time control applications, such as torque and speed regulation, because it can be implemented efficiently on standard computational platforms.
- *MEC* provides a middle ground between FEA and d-q EC, offering more detail than the d-q model while being less computationally demanding than FEA. MEC models can capture some of the non-linearities and magnetic interactions within the motor, making them useful for more detailed design analysis without the full computational load of FEA.

For the PMSM in this study, the d-q EC method was selected as the primary modeling technique. This choice was driven by the need for a model that could be used in real-time applications while maintaining high accuracy for dynamic performance analysis.

4.3.1 Advanced Mathematical Model

The dynamic equations in the stator and rotor reference frames must be considered to create a comprehensive and advanced d-q model for the PMSM. This approach encompasses the motor's electrical and mechanical dynamics [74]. Figure 7 shows the PMSM d-q equivalent circuit.

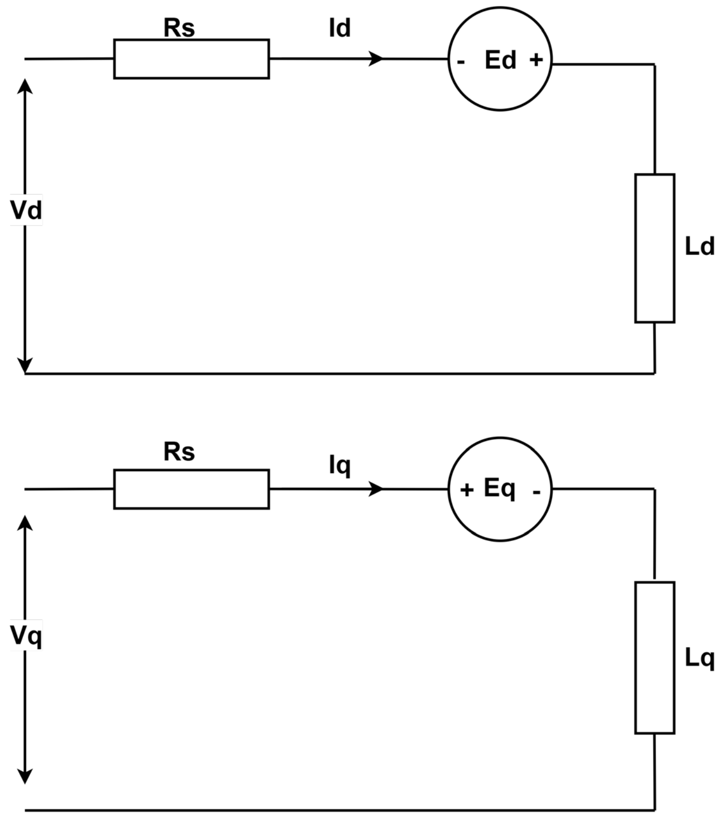


Figure 7. PMSM d-q equivalent circuit.

Detailed derivations are provided to achieve high-fidelity modelling, capturing the complexities of transient behavior and the interactions between the motor's various components.

- **Stator reference frame equations**

$$V_a = R_s i_a + \frac{d\lambda_a}{dt} \tag{6}$$

$$V_b = R_s\, i_b + \frac{\mathrm{d}\lambda_\mathrm{b}}{\mathrm{dt}} \tag{7}$$

$$V_c = R_s\, i_c + \frac{\mathrm{d}\lambda_\mathrm{c}}{\mathrm{dt}} \tag{8}$$

where $(V_a, V_b, V_c)$are the phase voltages, (i_a, i_b, i_c) are the phase currents, and $(\lambda_a, \lambda_b, \lambda_c)$ are the components of flux linkage.

- The flux linkage components are defined as

$$\lambda_a = L_s i_a + L_m\left(i_{m_a} + i_{m_b} + i_{m_c}\right) \tag{9}$$

$$\lambda_b = L_s i_b + L_m(i_{m_a} + i_{m_b} + i_{m_c}) \tag{10}$$

$$\lambda_c = L_s i_c + L_m(i_{m_a} + i_{m_b} + i_{m_c}) \tag{11}$$

where (L_s) is the stator self-inductance, (L_m) is the mutual inductance, and $(i_{m_a}, i_{m_b}, i_{m_c})$ are the magnetizing currents.

- **Rotor d-q reference frame**

The transformation from the *abc* to *d-q* reference frame is performed using the Park transformation:

$$V_d = \frac{2}{3}\left(V_a \cos\theta + V_b \cos\left(\theta - \frac{2\pi}{3}\right) + V_c \cos\left(\theta + \frac{2\pi}{3}\right)\right), \tag{12}$$

$$V_q = \frac{2}{3}\left(-V_a \sin\theta - V_b \sin\left(\theta - \frac{2\pi}{3}\right) - V_c \sin\left(\theta + \frac{2\pi}{3}\right)\right), \tag{13}$$

$$V_0 = \frac{1}{3}(V_a + V_b + V_c), \tag{14}$$

and

$$i_d = \frac{2}{3}\left(i_a \cos\theta + i_b \cos\left(\theta - \frac{2\pi}{3}\right) + i_c \cos\left(\theta + \frac{2\pi}{3}\right)\right), \tag{15}$$

$$i_q = \frac{2}{3}\left(-i_a \sin\theta - i_b \sin\left(\theta - \frac{2\pi}{3}\right) - i_c \sin\left(\theta + \frac{2\pi}{3}\right)\right), \tag{16}$$

$$i_0 = \frac{1}{3}(i_a + i_b + i_c), \tag{17}$$

- **Dynamic equations in d-q frame**

The dynamic equations in the dq0 frame, considering both the stator and rotor reference frames, are:

$$V_d = R_s i_d + \frac{d\lambda_d}{dt} - \omega_e \lambda_q \tag{18}$$

$$V_q = R_s i_q + \frac{d\lambda_q}{dt} + \omega_e \lambda_d \tag{19}$$

$$V_0 = R_s i_0 + \frac{d\lambda_0}{dt} \tag{20}$$

where:

$$\lambda_d = L_d i_d + \lambda_m, \tag{21}$$

$$\lambda_q = L_q i_q, \tag{22}$$

$$\lambda_0 = L_0 i_0. \tag{23}$$

Here, λ_m is the permanent magnet flux linkage, λ_d λ_q are the d-axis and q-axis inductances, respectively, andL_0 is the zero-sequence inductance.

- **Electromagnetic torque**

The electromagnetic torque (T_e) generated by the PMSM can be expressed as:

$$T_e = \frac{3}{2}p(\lambda_d i_q - \lambda_q i_d) = \frac{3}{2}p[\lambda_m i_q + (L_d - L_q)i_d i_q], \tag{24}$$

where p is the number of pole pairs.

- **Mechanical Dynamics**

The mechanical dynamics of the rotor are governed by:

$$J\frac{d\omega_m}{dt} = T_e - T_L - B\omega_m, \tag{25}$$

where:
J is the rotor's moment of inertia, B is the viscous friction coefficient, ω_m is the mechanical angular velocity, T_L is the load torque.

The power losses can be modelled as:

$$P_{loss} = P_{cu} + P_{core}, \tag{26}$$

$$P_{cu} = 3.R_s.(i_d^2 + i_q^2), \tag{27}$$

$$P_{core} = k_{core}.(\omega_e - \omega_{base})^2, \tag{28}$$

The above equations capture the complex dynamics of the PMSM, encompassing the interactions between the stator and rotor, the production of electromagnetic torque, and mechanical response.

4.3.2 PMSM Digital Model

The PMSM was represented digitally in the Simulink platform based on the advanced mathematical d-q model. The developed digital model comprises two interconnected blocks: electromagnetic, and mechanical. Each block plays a critical role in stimulating the motor's performance, reflecting its core dynamics. Figure 8 shows the PMSM digital model.

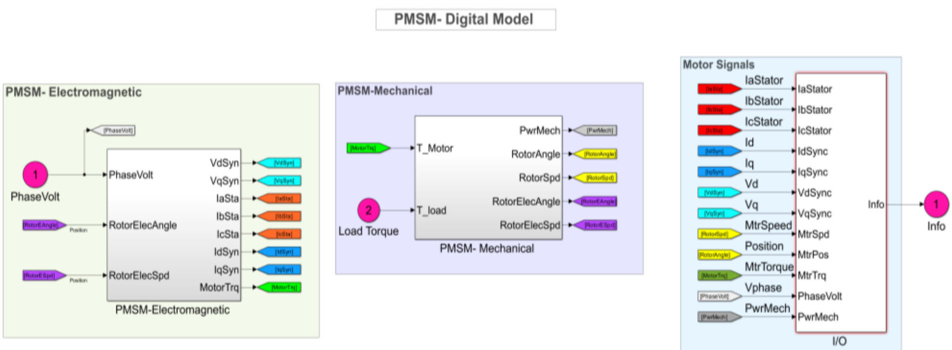


Figure 8. PMSM digital model [Paper V].

The PMSM electromagnetic block processes the motor's electrical inputs and calculates the resulting current responses. The process begins with the input stator phase voltages, which are initially in the three-phase ABC form. To simplify the calculations, a Clarke transform is applied, converting these three-phase voltages into the α-β stator reference frame. Following this, the Park transform is applied to convert the α-β components into the rotating d-q reference frame. The d-q frame moves with the rotor and allows for more intuitive manipulation of the motor's internal electromagnetic dynamics, such as flux and torque. Once the voltages are transformed, the PMSM equivalent circuit model is employed to compute the motor's reaction currents. This step simulates how the motor responds to the applied voltages, representing the flow of current through the stator windings. After calculating the currents in the d-q frame, the model reverses the transformation process. First, the inverse Park transform is applied to convert the d-q axis currents back into the α-β reference frame, and then the inverse Clarke transform is used to return these currents to the original three-phase ABC system. This completes the electrical cycle, bringing the currents back to their physical form for further use in motor control or analysis. Finally, the motor torque block uses the calculated stator currents and rotor position to determine the electromagnetic torque produced by the motor. This torque is a critical output, as it directly influences the motor's mechanical performance and is essential for determining the motor's speed and acceleration. Figure 9 shows the expanded subblocks of the PMSM electromagnetic block.

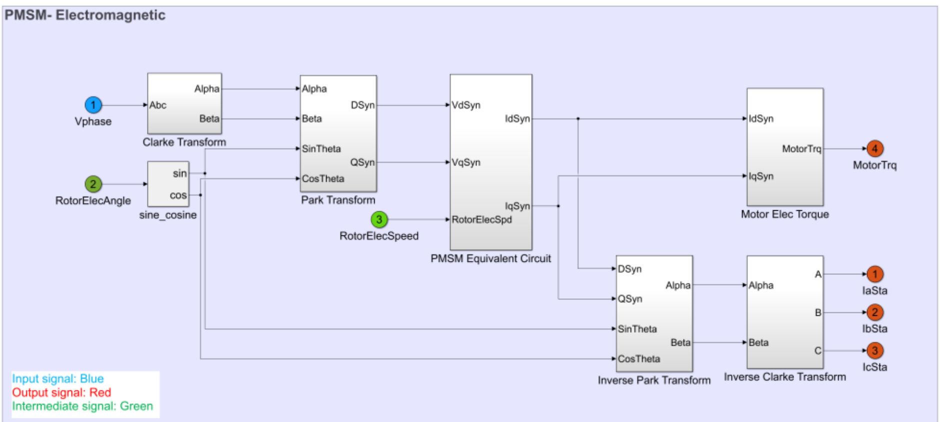


Figure 9. Expanded PMSM electromagnetic block [Paper V].

The mechanical block replicates the motor's rotational dynamics, accurately simulating the rotor's mechanical response to the electromagnetic torque generated by the motor. This block models key aspects of mechanical performance, including the rotor's speed, position, and inertia effects. The speed represents how fast the rotor turns in response to the torque, while the position tracks the angular displacement of the rotor over time. The inertia effects simulate the resistance to changes in motion, reflecting how the motor's mass and design influence acceleration and deceleration. By incorporating these elements, the mechanical block provides a realistic representation of the motor's mechanical behavior, crucial for accurately simulating performance in real-world applications.

The PMSM digital model is optimized to ensure compatibility with real-time execution on a target machine. These modifications include implementing a fixed-step solver (ode4) with a consistent time step of 100 µs, ensuring the model updates at regular intervals suitable for real-time operation. The fixed update rate allows the system to maintain synchronization with real-world processes, ensuring it can respond swiftly to dynamic changes such as variations in motor speed or load. These adjustments were critical for achieving the responsiveness and reliability required for real-time deployment.

4.4 PMSM Digital Model Validation

The next crucial step in the DT modelling process was to validate the developed digital model, ensuring it accurately mirrored the behavior of the physical PMSM under real-world conditions. To achieve this, test conditions were carefully selected to represent various modes of the EV driving cycle, reflecting real-world operational scenarios. This approach aimed to validate the virtual model under conditions that simulate different driving situations, such as acceleration, cruising, and deceleration, aligning closely with physical EV operation. Given that the vehicle's maximum speed is 20 km/h, equivalent to 1000 rpm for the motor, the drive system was configured to match this requirement. Multiple tests were conducted on the motor test bench under various loading and speed conditions, representing different stages of the driving cycle, as detailed in Table 5.

Table 5. Validation tests conditions [Paper V].

Case	% Load Torque	Speed (rpm)
A	80%	300
B	60%	500
C	40%	700
D	20%	1000

During the tests, the DAS captured key parameters such as motor phase current, phase voltage, and rotor speed. One key aspect of the testing setup was the method used to represent the applied torque. While no physical torque sensor was installed on the motor shaft due to the mechanical complexity of such an installation, the applied torque was approximated based on percentage torque values from the load emulators. This allowed for an indirect yet reliable means of assessing the loading conditions applied to the motor.

The acquiesced voltage, current, and speed data from the physical motor tests were reused in the digital model, ensuring that the physical and digital models were subjected to identical operating conditions for accurate comparison. The validation primarily focused on comparing the motor speed and current between the physical and digital models, as these were the parameters for which actual sensor data was available. The motor speed was precisely tracked using a resolver, and the current served not only as a direct validation parameter but also as an indicator of the loading torque. By validating against these key parameters, the digital model was confirmed to accurately reflect the physical motor's behavior. Figure 10 (a, b, c, and d) illustrates the validation results, showing the motor speed and stator current from both the actual measurements and the simulated outputs, demonstrating the accuracy of the digital model.

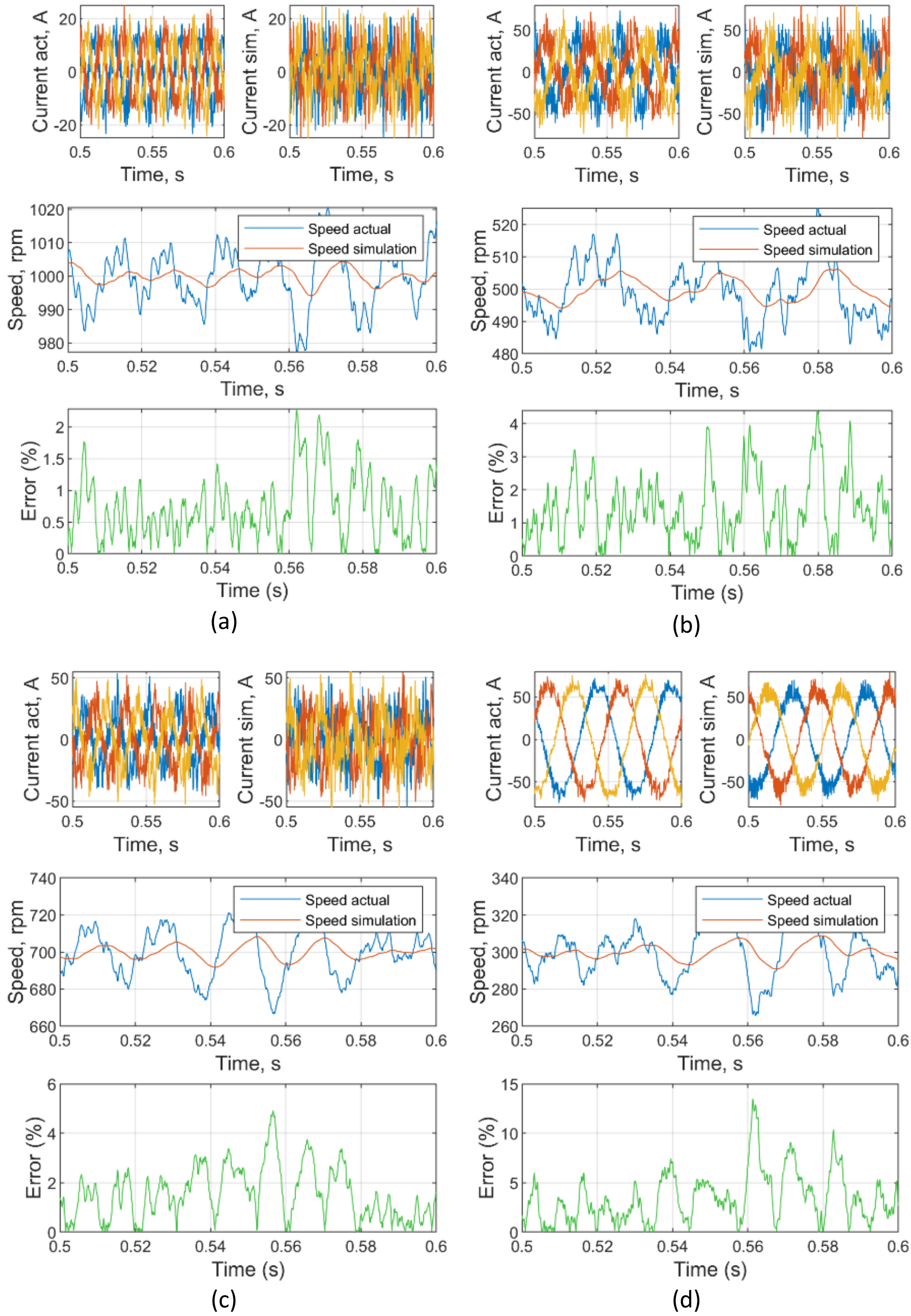


Figure 10. a-b-c-d Comparison between speed and stator current obtained from the physical and digital model [Paper V].

From the analysis of the above figures, several observations can be noticed. First, the speed from the digital model exhibited fewer ripples compared to the actual measured speed which can be attributed to the simulation environment does not account for all real-world factors such as sensor noise. The percentage error between the simulated and actual speed values remains within 5% across all cases, except in case d, where the error reaches up to 10% at certain points. This larger deviation is likely due to the motor operating at higher torque, which makes the system more sensitive to small variations in parameters such as resistance, and inductance. At higher speeds, the motor operates at lower torque, resulting in less sensitivity to parameter changes, which explains the lower error margin in these cases.

Both the actual and simulated stator currents exhibit not only closely matching peak values but also an identical number of pulses within a given time frame, indicating that the simulation accurately replicates the frequency characteristics of the actual system, capturing both the amplitude and temporal dynamics of the current signals. But its notable that the simulated stator current demonstrates higher accuracy at lower speeds (as observed in case D) compared to higher speeds (as in cases A and B). This can be explained as at lower speeds, the motor has more time to adjust the current to the desired value, resulting in better accuracy. However, at higher speeds, the reduced time for current adjustment leads to slight errors in current tracking. Notably, the RMS values of the simulation's steady-state stator currents closely match the measured values of the physical model, with no significant discrepancies.

4.5 Deployment to Digital Twin Framework for Performance Monitoring

The final phase in the DT development involved deploying the digital model to a DT framework to assess its computational efficiency, response time, and accuracy. To achieve this, the model was deployed to a real-time target machine a Speed goat Baseline, which acted as the DT in this setup. Complementing this, a host model was created to emulate the physical system. Both host and target models were equipped with nodes utilizing the UDP protocol to enable seamless communication and real-time data streaming between them.

To get the model subjected to diverse real-world conditions, a driving test was done on ISEAUTO, and the voltage, current, and speed data were collected using DAS from the EV motor during the test. The vehicle also had a dual GPS antenna system that tracks key performance metrics like velocity, distance, and altitude. This GPS data provides a comprehensive view of the vehicle's operation from different perspectives. The driving test was conducted over a 200-meter track at the university campus. Figure 11 illustrates the GPS data of the vehicle's velocity and altitude, while Figure 12 presents the physical EV motor voltage and current data captured during the test.

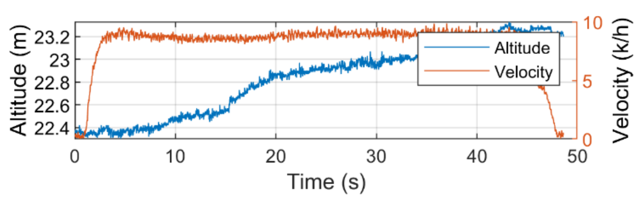


Figure 11. EV GPS-sensor data of vehicle velocity and altitude [Paper V].

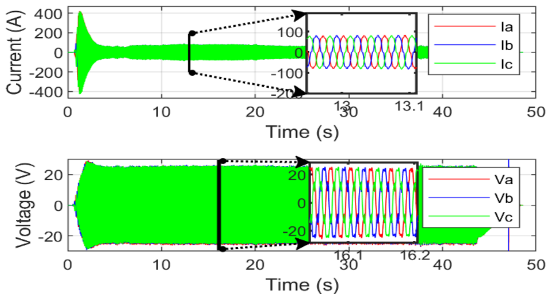


Figure 12. EV real-time sensor data of motor voltage and current during the test [Paper V].

The vehicle began from a standstill and gradually increased its speed until reaching a near-constant velocity. During this period, the altitude increased slightly from 22.4 m to 23.2 m above sea level, indicating a minor increase in the motor's applied torque. The vehicle's velocity can be converted to the corresponding motor speed in revolutions per minute (rpm) using the following equation:

$$v = 0.1885 \cdot N \cdot D \tag{29}$$

$w$here:
v – wheel velocity in km/h,
N – vehicle speed in rpm,
D – wheel diameter in m.

The collected data from the test was saved in MATLAB format to be used further in DT framework testing. The host model streamed the real-time voltage, and current to the target model. The target model processed this data, performed the necessary computations, and streamed out the speed and torque back to the host model. This bidirectional communication mimicked real-world operations, allowing for the comprehensive evaluation of the DT's performance and suitability. The results validated the model's ability to handle real-time data streams accurately and efficiently, confirming its readiness for real-time performance monitoring and predictive maintenance. Figure 13 displays the motor's performance metrics from the DT model side, while Figure 14 compares the velocity estimated by the DT with the actual velocity measured via GPS.

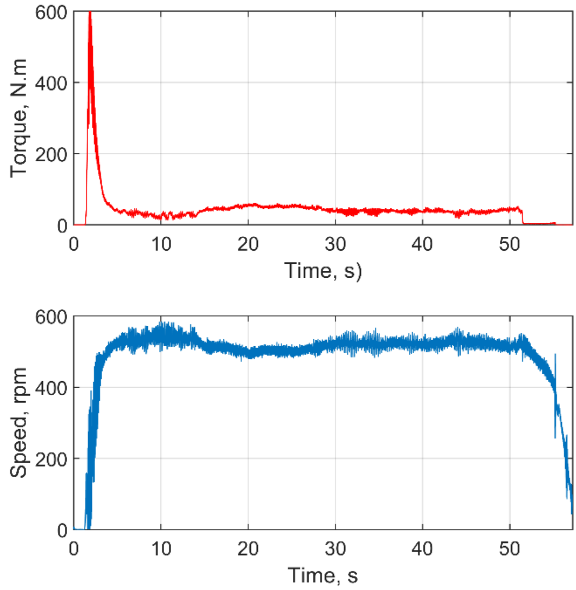


Figure 13. EV motor torque and speed from the DT model [Paper V].

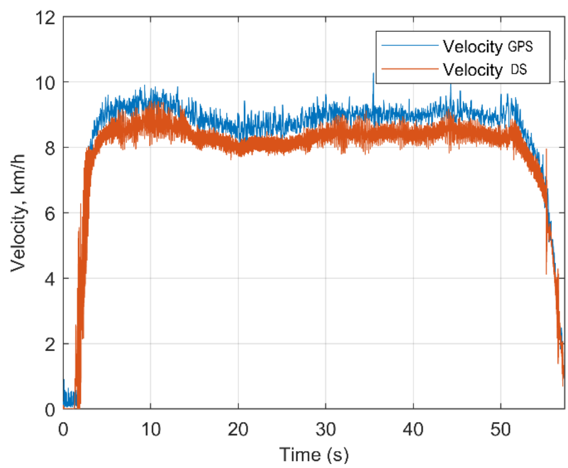


Figure 14. Vehicle Velocity from GPS and DT model [Paper V].

The analysis of Figure 13 demonstrates a strong correlation between the motor's torque and current behavior, as shown in Figure 12. The data indicates that the torque required to move the vehicle from a standstill is significantly high, which is reflected by the corresponding surge in starting current. Between 10–23 s, the motor torque increases while there is a slight decrease in speed, indicating the vehicle is accelerating under load – this is consistent with the slight rise in altitude observed in Figure 11. From 23–30 s, the speed gradually increases to the nominal test speed as the torque normalizes, showcasing the motor's ability to adapt to varying driving conditions.

Figure 14 shows a high level of agreement between the GPS-measured velocity and the velocity predicted by the DT, with a percentage error of less than 3%. This low error rate highlights the accuracy of the DT in monitoring system performance. The error margin can be attributed to the inherent limitations of GPS accuracy. Factors such as the number of satellites, signal quality, and the type of GPS receiver used can influence the precision of GPS measurements.

During the real-time testing phase, the DT model demonstrated practical computational performance suitable for real-time applications. It utilized approximately 55% of the CPU capacity, indicating a moderate demand on processing resources, leaving some headroom for additional tasks or future enhancements. Memory usage was around 60% of the available RAM, which is acceptable for real-time operation. The response time of the model was consistently under 2 ms per simulation step, ensuring that the system could process incoming data and generate outputs with minimal latency.

4.6 Chapter Summary

This chapter presented the DT modeling framework of the EV-PMSM, providing a comprehensive methodology. The work began with collecting all available data about the physical motor model, ensuring a solid foundation for the modeling process. Based on this data, the appropriate representation method was selected, starting with model identification and parameter estimation, where the ALO algorithm was employed to estimate the motor's lumped parameters. Following this, the d-q mathematical model was chosen as the foundation for the motor's digital representation due to its ability to accurately capture the motor's dynamics. Using MATLAB/Simulink, the digital model was created with key features, including the ability to perform real-time simulations, which is critical for the functionality of the DT. After the model was developed, it was validated against the physical motor model, ensuring that the digital model accurately mirrored real-world behavior under various operational conditions. The final step involved deploying the digital model within a DT framework, where sensory data from the EV motor was streamed in real time to the digital model deployed to a real-time target machine. This allowed the DT to simulate the motor's output torque and speed in real time, demonstrating its effectiveness in reflecting its performance and condition. Figure 15. Illustrates the framework used to develop the motor DT.

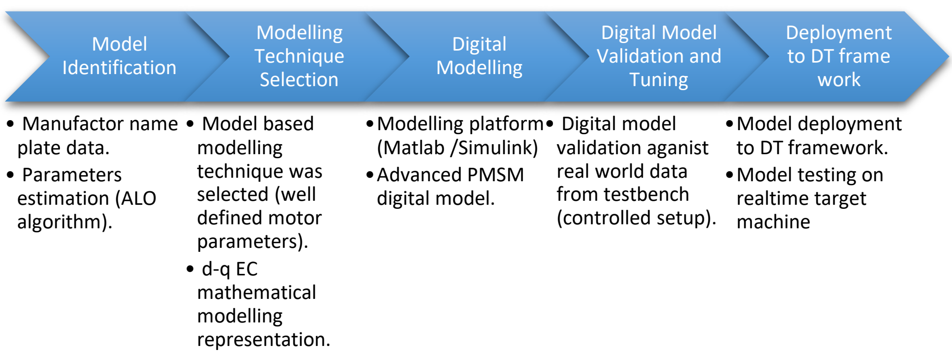


Figure 15. Development framework of the EV motor DT model.

Although the DT demonstrated strong performance in replicating motor behaviour, some areas require further attention. Specifically, the indirect methods used for torque estimation could benefit from further refinement to improve accuracy. Additionally, testing under more extreme operating conditions, such as rapid acceleration or changes in terrain, would enhance the robustness of the DT. The chapter successfully demonstrated the potential of the DT for real-time monitoring and condition assessment of the EV's PMSM, while also highlighting areas for further optimization.

5 EV-Powertrain Digital Twin Modelling Framework

The powertrain is the central component of the EV, responsible for converting electrical energy from the battery into mechanical motion and ultimately defining the vehicle's overall efficiency. It consists of two key elements: the motor and the traction drive system. The motor drive system is critical in shaping the electrical inputs from the battery into the signals necessary to achieve the desired mechanical output from the motor. It controls the motor's performance and optimizes energy usage, ensuring the efficiency of the system.

This chapter outlines the development framework of the DT model for the EV powertrain, combining both the motor and drive system covering the key work presented in research papers [VI], [VII], and [VIII]. Unlike the motor DT developed in the previous chapter using a model-based approach, the motor drive system presents greater challenges due to its complexity and the higher degree of uncertainty in key parameters, particularly the control strategy. These complexities made it impractical to rely solely on a model-based framework, necessitating the adoption of an advanced modeling methodology for the traction drive system's DT development.

To address this complexity, a data-driven approach is employed for modeling the traction drive system. By conducting extensive testing on the testbench, input-output data from the physical system are collected under various operating conditions. AI algorithms are utilized to capture the system's dynamics based on this empirical data. This approach offers several benefits. Adaptability is achieved as the model can learn complex nonlinear relationships without explicit knowledge of the internal control algorithms. Accuracy is enhanced by training on real-world data, allowing the model to replicate the actual performance of the drive system closely.

A comprehensive hybrid-driven powertrain DT model is achieved by integrating the data-driven model with the previously developed model-based motor model. This hybrid model combines the strengths of both data-driven and physics-based approaches, providing a high-performance, reduced-order representation of the EV powertrain.

5.1 Motor Drive System Model Identification

Identifying the motor drive system model is critical in digitally representing the system. The used traction drive has the following specifications presented in Table 6.

Table 6. Drive system specifications.

Component	Specification
Power Rating	55 kW
Current Rating	350 A
Input Voltage	400–750 VDC
Output Voltage	0–500 VAC
Cooling	Liquid-cooled (Water-Glycol mixture 50-50)
Ambient Temperature	-40 °C to +85 °C
Protection Rating	IP67 (dust and waterproof)
Controller	Vector control with flux weakening
Programming Standard	IEC 61131-3

The drive employs a vector control strategy, which is essential for achieving precise and efficient control of the motor's torque and flux – key factors in optimizing the performance of EV powertrain. Vector control allows independent regulation of these parameters, enabling the motor to respond effectively to varying load conditions and speed requirements.

Vector control operates by transforming the motor's three-phase stator currents into a two-axis coordinate system, simplifying the complex dynamics of PMSM control. This process begins with converting the three-phase stator currents into two orthogonal components in a stationary reference frame (α-β). This transformation reduces the three-phase system to an equivalent two-axis system while retaining all the necessary information about the motor's magnetic state. Next, these components are transformed into d-q, rotating a reference frame that is aligned with the rotor's magnetic field. By aligning the reference frame with the rotor flux, the current components independently control the magnetic flux and torque production. This decoupling allows for separate control of flux and torque.

Once transformed, the motor control system can adjust the flux and torque-producing currents independently using controllers, typically involving proportional-integral (PI) controllers, to meet the desired performance criteria. For instance, the torque-producing current is adjusted based on the torque demand, while the flux-producing current is managed to maintain the optimal flux level. After computing the required current components, inverse transformations are applied to convert these quantities back to the three-phase system for implementation. These transformations yield the three-phase current references that are used to generate the gating signals for the IGBT transistors in the inverter. The inverter then modulates the voltage applied to the motor windings accordingly. However, accurately modeling the motor drive system presents challenges due to the uncertainty of specific control parameters and the lack of detailed documentation on the control algorithm used. Key parameters such as controller gains, current and voltage limits, and the exact implementation details of the control strategy are not fully disclosed.

5.2 Data Acquisition and Preprocessing

5.2.1 Data Collection and Gathering

To thoroughly assess the performance of the EV motor drive system, comprehensive tests were conducted on the test bench under controlled and diverse conditions. The test bench is equipped with an extensive array of sensors – including voltage, current, and speed sensors – all directly connected to the DAS. Tests were meticulously designed to emulate the EV motor drive system performance under realistic operating conditions. These tests simulated both the constant torque (CT) and constant power (CP) regions characteristic of the ideal EV performance profile, as depicted in Figure 16.

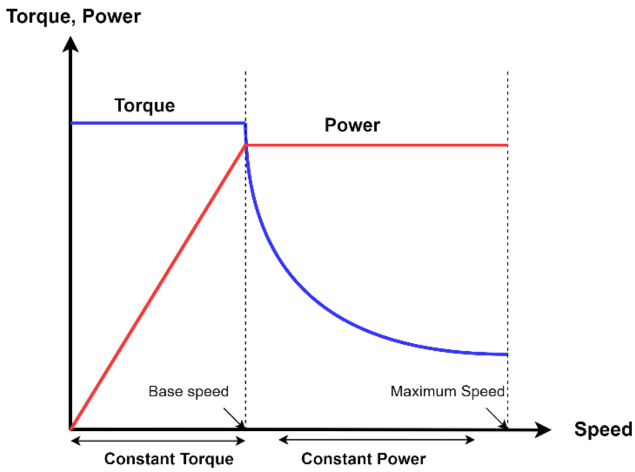


Figure 16. Torque, power/speed profile of ideal EV-PMSM drive system [70].

In designing the tests, the torque was initiated at nearly its rated value and then decreased as the speed increased. This approach ensured that the test cases covered a wide spectrum of operational scenarios – from low-speed, high-torque conditions to high-speed, low-torque conditions – effectively mirroring the EV's performance across its entire operating range. Table 7 presents the reference values used in the tests to achieve the desired outputs.

Table 7. Data collection Tests conditions [75].

Test	Ref. Speed (p.u)	Ref. Load Torque (p.u)
a	0.1	1.00
b	0.2	1.00
c	0.3	1.00
d	0.4	0.68
e	0.5	0.49
f	0.6	0.31
g	0.7	0.27
h	0.8	0.23
i	0.9	0.18
J	1.0	0.12

This comprehensive testing strategy was pivotal in capturing a broad dataset that reflects the dynamic interplay between speed and torque in the EV drive system. The collected data from these tests serve as critical input for the data-driven modeling process, enabling the construction of a surrogate model that accurately represents the drive system's response to varying operational conditions. The dataset obtained is characterized by precise collection methods, notably featuring a high sampling frequency of 10 kHz. This high-frequency data acquisition ensures a remarkable level of accuracy, capturing the intricate dynamics and transient behaviour of the drive system with precision.

5.2.2 Data Preprocessing

The second step after data collection involves thorough data preprocessing, a critical phase to ensure high-quality, reliable data for effective modeling. All preprocessing steps in this study were performed using MATLAB, leveraging built-in functions to refine, clean, and analyze the collected data, thereby extracting meaningful insights for accurate modeling.

Noise removal was the first step, targeting irrelevant fluctuations and measurement errors in the raw data. Using MATLAB's filtfilt function, a zero-phase Butterworth low-pass filter was applied to smooth high-frequency fluctuations, ensuring that only meaningful variations reflecting the system's behavior were retained without introducing phase distortion. Next, missing values were handled through linear interpolation, a technique that estimates missing data points based on adjacent values. This was implemented using the fillmissing function with the 'linear' option, generating a continuous dataset that preserves the data's inherent structure while maintaining consistency.

Outlier detection and removal were performed using the Interquartile Range (IQR) method. MATLAB's filloutliers function with the 'quartiles' method identified data points lying outside 1.5 times the IQR from the lower or upper quartiles. These flagged outliers were systematically removed, mitigating the impact of extreme values on model accuracy.

For feature scaling, standardization was applied to ensure all features had a mean of zero and a standard deviation of one, using MATLAB's normalize function with the 'zscore' method. This step ensures that each feature contributes equally to the model, avoiding disproportionate influence from larger-scale variables. Table 8. summarize the utilized MATLAB functions for the data preprocessing.

Table 8. Summary of the used preprocessing MATLAB functions.

Step	MATLAB Function	Purpose
Noise Removal	filtfilt	Apply a zero-phase Butterworth low-pass filter to remove high-frequency noise without introducing phase distortion.
Missing Data Handling	fillmissing	Perform linear interpolation to estimate and fill missing data points.
Outlier Detection	filloutliers	Identify and remove outliers using the Interquartile Range method.
Feature Scaling	normalize	Standardize features to have a mean of zero and a standard deviation of one.

An emphasis was placed on understanding the relationships between various operational parameters through correlation analysis. Continuous Analysis of Variance (CANOVA) was utilized as a statistical method to achieve this. Unlike traditional correlation analysis, CANOVA is adept at handling complex datasets where the goal is to assess the impact of continuous variables on the system's performance. This approach is particularly useful in scenarios where multiple input variables might influence a range of output variables in a nonlinear and interactive manner. CANOVA operates on the principle of analyzing the variance in the data that can be attributed to different factors and their interactions, providing a framework to understand how changes in input variables – like reference speed and torque – continuously affect outputs such as actual speed, torque, voltage, and current. The general approach in CANOVA involves fitting a model that describes the relationship between inputs and outputs. For instance, considering a simplified scenario with one input X and one output Y the model could be represented as:

$$Y = \beta_0 + \beta_1 X + \epsilon \tag{30}$$

where,
Y represents the output variable, X represents the input variable, β_0, β_1 are coefficients that describe the intercept and the slope of the relationship, respectively, ϵ represents the error term, accounting for variability not explained by the model.

After analyzing the inherent correlations between key operational parameters, clear relationships were observed between current and actual torque, as well as between actual speed and voltage. These correlations, deeply rooted in the motor's electromagnetic and electrical characteristics, suggest that the electrical current is a direct predictor of the motor's torque output, while the applied voltage and frequency dictate the motor's operational speed. Leveraging these relationships allows for a significant reduction in the complexity of the model's inputs.

5.3 Motor Drive System Initial Model

In the initial modeling phase, as previously discussed, an elementary approach was adopted by considering the drive system as a single unified block, despite comprising two main components, the inverter and the controller. This simplification allowed for the development of a more generalized model, serving as an initial indication of the success of the proposed data-driven methodology without delving into the complexities of each component.

The elementary model was developed using a deep ANN (DANN) implemented within the MATLAB environment, employing a nonlinear regression approach. The following equation can mathematically represent this approach:

$$K_i = h\left[x_i(1), x_i(2), \dots, x_i(m) \,;\, \sigma_1, \sigma_2, \dots, \sigma_p\right] + \epsilon \tag{31}$$

where,
K – number of responses, h – the function, x – inputs, σ – parameters being estimated, and ϵ – error term.

The DANN can recognize complex data patterns by utilizing multiple layers of interconnected neurons. Each layer extracts different features from the input data, allowing the network to capture both linear and nonlinear relationships inherent in the system. Information flows from one layer to the next in a hierarchical manner, enabling the network to learn intricate representations through the deep learning process.
The network was trained using the backpropagation algorithm, a widely adopted method for training feedforward neural networks. A learning rate of 0.01 was selected to control the step size during the weight updates, ensuring effective learning without causing instability. A momentum term of 0.9 was incorporated to accelerate convergence and help the algorithm avoid local minima by smoothing out the updates based on past gradients. These hyperparameters were chosen based on empirical testing to achieve a balance between learning speed and model stability. The dataset was divided into training and testing sets. (80%) of the data was used for training, allowing the network to learn the underlying patterns and relationships within the data. The remaining (20%) was reserved for testing, providing a means to evaluate the model's accuracy on unseen

data. Specifically, test cases a, b, d, e, g, h, and j were used for the training process, exposing the model to a diverse range of operating conditions. The rest of the test cases – c, e, and j – were kept for the testing process. This split ensured that the model's ability to generalize to new inputs could be effectively assessed, reducing the likelihood of overfitting. Figure 17 illustrates the architecture of the ANN employed in this modelling approach.

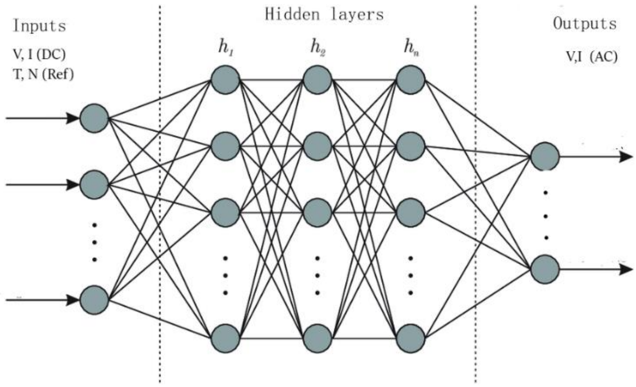


Figure 17. DANN architecture used for elementary data-driven model [Paper VI].

The network consists of an input layer that receives the input variables from the dataset, multiple hidden layers composed of neurons that apply activation functions to the weighted inputs, enabling the network to model complex nonlinear relationships, and an output layer that produces the predicted outputs corresponding to the system's response.

The input data were organized into four main clusters (A, B, C, D), corresponding to the different test scenarios conducted during data collection. Each cluster comprised five sub-inputs representing various testing cases. Every sub-input included four key variables: DC voltage and current, reference speed, and torque. Similarly, the output data were grouped into four responses aligned with the testing cases, each comprising two sub-responses for AC voltage and current. The two sub-responses included six variables corresponding to the three-phase voltages and currents. To enhance the model's accuracy and provide multiple modeling options, three types of ANNs were employed in the model generation process: Wide ANN, Billiard ANN, and Optimized ANN.

The results indicated that the Billiard ANN achieved the highest accuracy among the three models, exhibiting the lowest Root Mean Squared Error (RMSE). However, the Optimized ANN provided better coverage of predicted points over the true values. Table 9 summarizes the training results of the three DANN configurations.

Table 9. Performance comparison between DANN configurations.

DANN Configuration	**RMSE**	**MAE**	**R^2 Score**
Wide DANN	0.1481	0.1185	0.893
Billiard DANN	0.1327	0.1062	0.951
Optimized DANN	0.1406	0.1125	0.927

Despite the lower RMSE of the Billiard ANN, its focus on accuracy led to the omission of some parameters, which limited the predicted output range. Consequently, when validation was conducted with all parameters included, the accuracy of the Billiard ANN decreased. Ultimately, the Optimized DANN was selected due to its balance between accuracy and comprehensive coverage of all parameters. The selected model was extracted as a compact MATLAB function capable of handling input datasets and effectively predicting the outputs. Figure 18 presents the results of the model predictions compared to the true values for the selected optimized DANN.

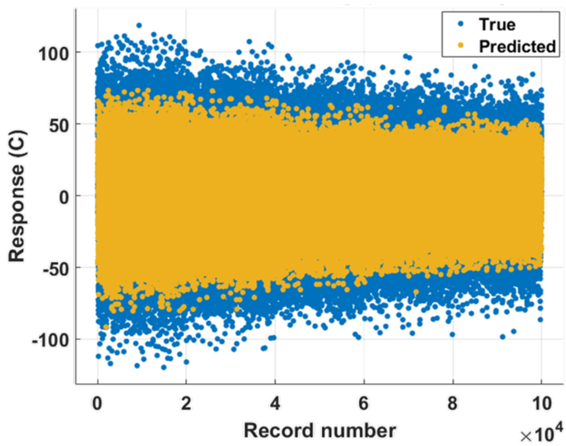


Figure 18. Training score of the Optimized ANN [Paper VI].

5.3.1 Initial Model Validation

The developed DANN model was validated through a two-step verification process to ensure its accuracy and reliability.

In the first verification step, the model was tested using the seen data – the same input data utilized during the training phase. This evaluation aimed to assess the model's ability to replicate known outcomes and verify that it had effectively learned the underlying patterns within the dataset. Figures 19 show a comparison between the actual and predicted outputs of case b as summarized from paper [VI].

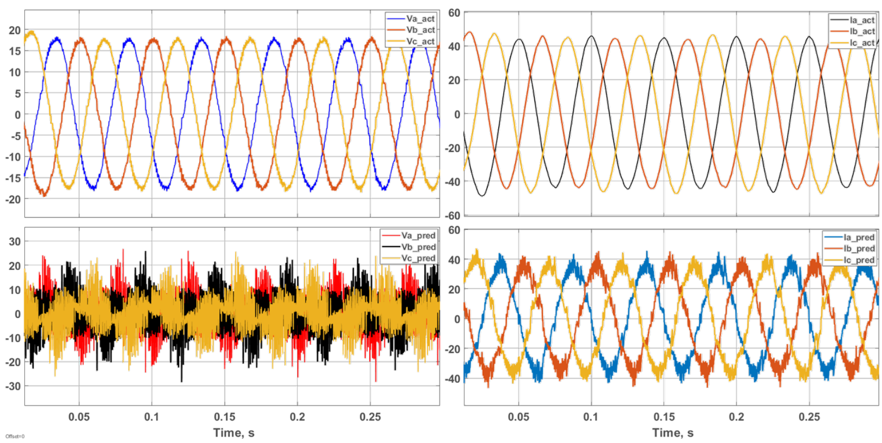


Figure 19. Actual vs predicted data, case b [Paper VI].

Upon analysis, it was observed that the predicted values from the DANN model exhibited more noise compared to the actual measured values. This is evident in the above Figures, where the fluctuations in the predicted data are more pronounced than those in the actual data. Despite the increased noise, the model demonstrated a reasonable level of accuracy, with the Root Mean Squared (RMS) values of the predicted and actual data being nearly close. The error was less than 10% in most cases, indicating that the model could effectively capture the overall trend and average behavior of the system.

In the second verification stage, the model's ability to generalize was assessed by testing it on a new dataset comprising input data from test cases not used during the training phase. This step was crucial for evaluating the model's performance in predicting responses to unseen conditions, which is a key indicator of its robustness and practical applicability. The input data for this stage were taken from test cases c and j, which were deliberately excluded from the training process to serve as a fresh benchmark. The model's predictions were then compared against the actual measured data from these test cases. Figures 20 and 21 show a comparison between the actual and predicted outputs of test cases c and j.

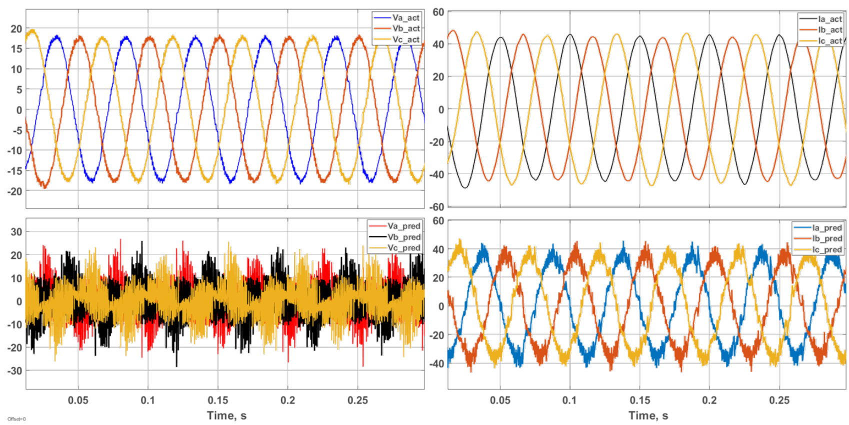


Figure 20. Actual vs predicted data, case c [Paper VI].

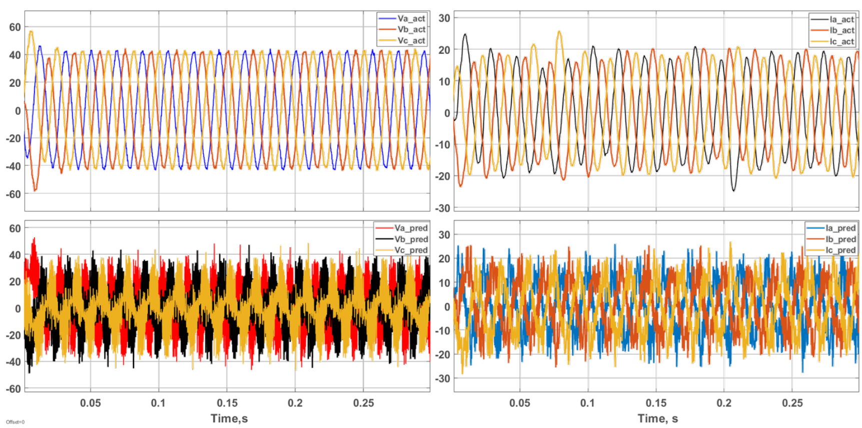


Figure 21. Actual vs predicted data, case j [Paper VI].

By analyzing the results from the above figures, it became apparent that the predictions still exhibited a significant amount of noise compared to the actual measured values. The figures illustrating the model's outputs versus the true values for these unseen test cases clearly show discrepancies, with the predicted data fluctuating more erratically than the actual system responses. This persistent noise indicates that the model struggles to generalize effectively to new operating conditions not encountered during training.

The presence of such noise in the predictions poses a challenge for practical application within the DT framework. This observation suggests that, while the elementary model provided initial insights and demonstrated the feasibility of a data-driven approach, it lacks the robustness required for deployment in a practical DT framework. This realization led to the development of a more advanced model using the Nonlinear Autoregressive Neural Network with Exogenous inputs (NARX), which is designed to handle temporal dependencies and nonlinearities more effectively, ultimately improving prediction accuracy and reliability for practical applications.

5.4 Motor Drive System Advanced Model

Building upon the initial model, the next phase involves developing a more detailed and accurate model of the drive system. This approach allows for a granular representation by dividing the drive system into two main components: the inverter and the controller, as illustrated in Figure 22. By modeling each component individually, this methodology offers significant advantages over the elementary model that considers the entire drive system as a single block. This not only simplifies the modeling process but also leads to more accurate and efficient simulations.

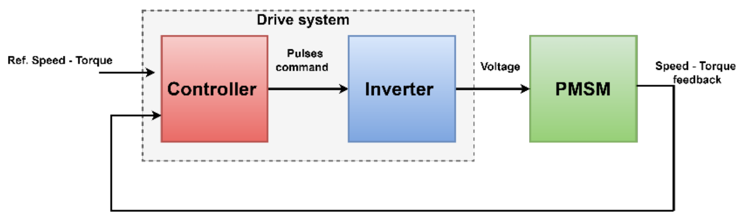


Figure 22. EV- powertrain model diagram [Paper VII].

5.4.1 Inverter Model

The inverter is a well-defined component with established electrical characteristics and operational principles. Its primary function is to convert DC power from the battery into AC power supplied to the motor, adjusting the voltage and frequency to control motor speed and torque. As well-known semiconductor device models govern the inverter's behavior, a model-based approach is appropriate for its representation. The inverter model was developed using an Average Value Inverter (AVI) block within the Simulink environment. The AVI simplifies the representation of the inverter by averaging the effects of the high-frequency switching actions of the power electronic devices, such as IGBTs or MOSFETs. Instead of modeling each switching event, the AVI provides an averaged effect of the inverter's output over a switching period, significantly reducing the computational burden while retaining the essential dynamic characteristics necessary for accurate system-level simulation.

This approach offers several advantages. By eliminating the need to simulate each high-frequency switching transient, the model enhances computational efficiency, making it particularly suitable for real-time simulations and control system development where computational resources are limited.

5.4.2 Controller Model

The controller in the EV drive system plays a crucial role in implementing the control strategy that dictates the motor's operation. It adjusts inputs such as voltage and current to achieve the desired speed and torque based on reference commands. Unlike the inverter, the controller's internal parameters and algorithms are complex and partially unknown which makes it challenging to be modeled using traditional model-based approaches.

The NARX model is particularly well-suited for modelling dynamic systems with feedback loops and temporal dependencies, characteristics inherent in the drive system. By incorporating past values of the output (autoregressive terms) and external inputs (exogenous inputs), the NARX model can capture complex nonlinear relationships and time-dependent behaviors more effectively than traditional feedforward neural networks.

The model primarily receives reference torque and speed as inputs, which serve as directives to the actual drive system to achieve the desired operational states. The current and voltage represent the immediate control actions executed by the system, while the feedback speed and torque reflect the motor's actual response to these control signals. To accurately capture the drive system's behavior – including its response delays and the effects of feedback loops – the NARX model integrates both the reference inputs (torque and speed) and the feedback outputs (torque and speed) into its input set, applying the feedback inputs with a delay. This delay is crucial for modeling the temporal dynamics of the system, reflecting the time it takes for control actions to impact the motor's operation and for these effects to be observed through feedback mechanisms.

The model operates as follows:

- Immediate Inputs: Reference torque and speed at time t, which represent the target operational parameters set by the drive system.
- Delayed Feedback Inputs: Feedback torque and speed from a previous timestep ($t-\delta$) where δ represents the delay that accounts for the system's response time. This inclusion of delayed feedback inputs allows the model to learn from the system's past responses to adjust its future outputs, mirroring the real-world feedback loop in the control strategy.

The outputs of the model are designed to predict:

- Current and Voltage: The necessary electrical inputs to the motor that result from processing the reference and feedback inputs, aiming to achieve the reference operational states.
- Feedback Speed and Torque: These are also modeled as outputs to evaluate the model's accuracy in predicting the system's actual response, providing a comprehensive view of the control strategy's effectiveness.

5.4.3 NARX Architecture

The NARX model is designed with a layered architecture, beginning with an input layer configured to align with the dimensions of both the reference inputs and the delayed feedback signals. Central to the design are three Gated Recurrent Unit (GRU) layers, interspersed with two dropout layers, which work together to enhance the model's generalization capabilities and mitigate overfitting. The GRU layers are integral to the model, selected for their exceptional ability to manage sequential data and retain information over extended periods. This feature is particularly critical for capturing the dynamics of feedback loops and the delayed effects characteristic of the EV drive system's control strategies. Each GRU layer is equipped with 128 neurons, a configuration that balances computational efficiency with the ability to learn complex patterns and relationships in the data. GRUs were specifically chosen over other recurrent units due to their simplified architecture, which maintains the capacity to capture long-term dependencies with fewer parameters, thereby streamlining the training process. Between the GRU layers are two dropout layers that act as a regularization mechanism. During training, these layers randomly deactivate a subset of neuron connections, reducing model complexity and helping to prevent overfitting. This approach encourages the model to develop more generalized features by reducing its reliance on specific inputs or patterns. The dropout rate, typically ranging from 0.2 to 0.5, is determined based on empirical testing and validation performance. For optimization, the model leverages the Adam optimizer, recognized for its adaptive learning rate mechanism. This optimizer efficiently navigates the intricate landscape of time-series data by adjusting learning rates dynamically, utilizing the first and second moments of the gradients. This adaptability accelerates convergence toward optimal solutions while maintaining stability in the training process. Finally, a sigmoid activation function is employed within the GRU layers to introduce non-linearity, further enhancing the model's ability to capture complex relationships and dependencies in the data. The sigmoid function is defined as:

$$f(x) = \frac{1}{1 - e^{-x}} \tag{32}$$

The MSE is employed as the loss function to measure the discrepancy between the predicted outputs and the actual observations. This choice quantifies the average squared differences between the predicted and true values, guiding the model's learning process toward minimizing these errors and enhancing predictive accuracy. An illustrative diagram of the NARX architecture is presented in Figure 23.

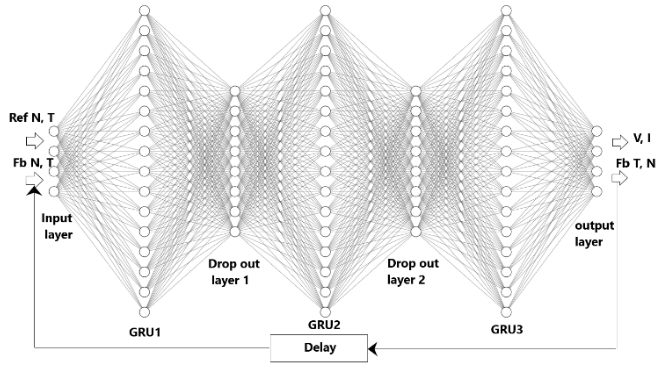


Figure 23. NARX architecture.

5.4.4 NARX Model Training

The NARX model was developed using MATLAB's Deep Learning Toolbox through a meticulous and methodical process to ensure a robust predictive capability for the EV drive system's behavior. The test bench data were strategically partitioned to optimize learning and evaluation, with the dataset divided into training data and unseen data, each serving a specific purpose in assessing the model's performance. The training data constituted the majority of the dataset and included cases a, c, d, e, f, g, and j, chosen to represent a broad spectrum of operational scenarios for effective learning. In contrast, cases b, h, and i were reserved as unseen data and excluded from the training and validation phases to provide a stringent evaluation of the model's generalization ability. These unseen cases were selected strategically to encompass varying operational conditions, including both CT and CP regions, ensuring a comprehensive assessment of the model's predictive capabilities.

The training process involved feeding the training data through the NARX model, during which weights and biases were iteratively updated using MATLAB's train function with the Levenberg-Marquardt optimization algorithm. This optimization approach was selected for its ability to handle non-linear datasets efficiently, a critical requirement given the intricate dynamics of the EV drive system. Approximately 75% of the dataset was allocated for training, with 15% reserved for validation to monitor performance and prevent overfitting, and the remaining 10% held back for testing to evaluate the model on unseen data. Hyperparameters were carefully tuned to optimize model performance, including an initial learning rate of 0.001, which was adjusted dynamically during training to ensure convergence without overshooting, and a training duration of 500 epochs, which balanced performance improvement with computational efficiency. The model training was conducted over 27 hours on a high-performance workstation equipped with an Intel Core i9-10900X processor (3.7 GHz base frequency, 10 cores, and 20 threads) and 64 GB of DDR4 RAM. This setup provided the computational power necessary for handling the extensive dataset and the demands of the NARX model's architecture. Throughout the training, validation performance was continuously monitored every 10 epochs to detect signs of overfitting early. Early stopping criteria were implemented, halting training when validation loss plateaued over multiple epochs, thereby avoiding unnecessary computations. The model's performance was evaluated using the MSE loss function, which was minimized during training to ensure that predictions closely matched actual system behaviour. The validation set served as an essential measure of the model's generalization capabilities, while the final testing phase provided a robust assessment of accuracy on unseen data. As illustrated in Figure 24, the model demonstrated a consistent reduction in MSE over epochs, showcasing its ability to capture complex patterns in the data and converge toward an optimal solution, highlighting the efficacy of the training process in developing a reliable predictive model for the EV drive system.

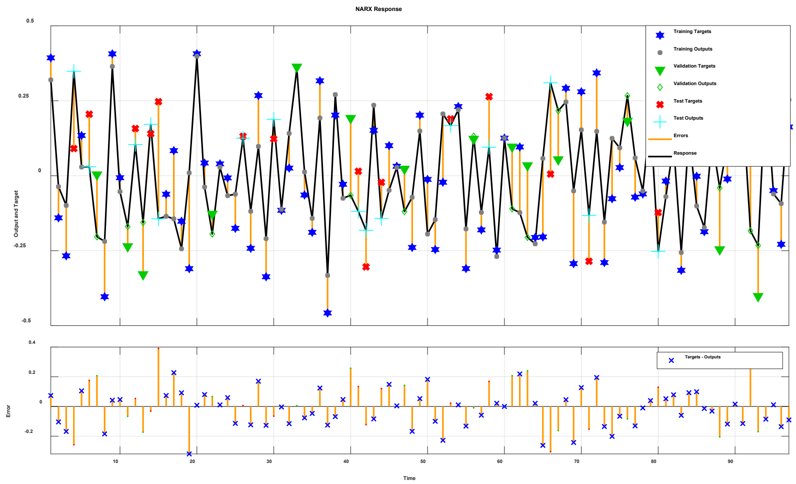


Figure 24. NARX training performance.

The performance metrics of the NARX model, reflected in the mean squared error (MSE) values across the training, validation, and testing phases, illustrate its exceptional capability to model the dynamics of the EV PMSM drive system. The model achieved an MSE of 0.0126 during training, demonstrating a strong capacity to learn from the provided data. This performance was further supported by the validation phase, where the model attained an MSE of 0.0214, highlighting its ability to generalize effectively to data it was not explicitly trained on but encountered in a similar context. The ultimate measure of the model's efficacy, however, lies in its performance on completely unseen data, where it achieved an MSE of 0.0406. While this represents a modest increase compared to the training and validation phases, the error remains remarkably low, emphasizing the model's robustness and its high accuracy in predicting system behavior under new conditions. The gradual increase in MSE from training to testing is a common and expected outcome in machine learning, as it reflects the model's effort to generalize beyond the specific patterns in the training dataset without overfitting. Furthermore, the relatively small differences between the MSE values across the training, validation, and testing phases underscore the model's stability and reliability. This consistency across datasets indicates a well-balanced model, capable of accurately capturing the complex dynamics of the EV PMSM drive system while maintaining robust performance under varied and previously unseen scenarios.

5.5 EV Powertrain Hybrid Model

After completing the training and validation phases, the NARX model was exported as a Simulink block to facilitate a seamless combination with the PMSM digital model forming a hybrid-driven model. Figure 25 illustrates the hybrid model of the EV powertrain system.

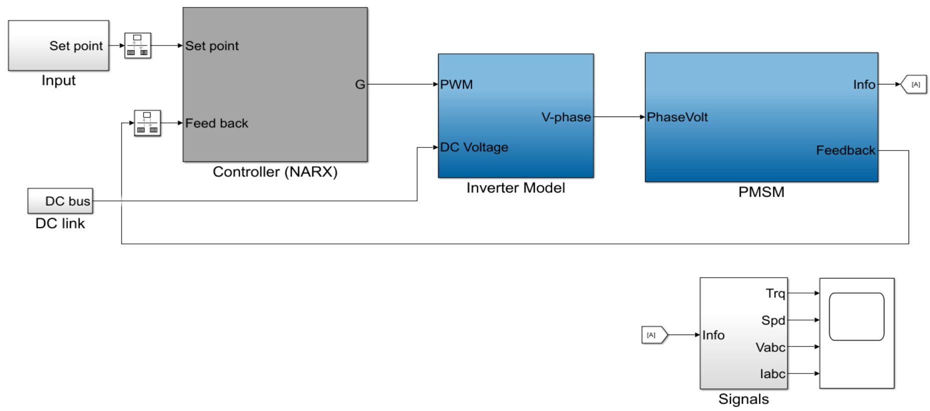


Figure 25. EV-power train hybrid driven model.

The hybrid model comprises four essential blocks that synergistically simulate the system's dynamics with high fidelity while maintaining computational efficiency. The first block is the Reference Input block, which provides the driving scenario by supplying operational setpoints such as desired motor speed and torque profiles. The second block is the data-driven controller model, implemented using the NARX. This model captures the controller's behavior by receiving reference inputs from the reference block and feedback signals from the motor of actual speed and torque. The NARX model processes these inputs to generate the necessary PWM commands. By incorporating both current and delayed inputs, it effectively captures the temporal dynamics and nonlinearities inherent in the control strategy of the EV drive system. The use of a data-driven NARX model provides a reduced-order representation of the controller, enabling high-speed computation without compromising the accuracy of dynamic behaviors, which is critical for real-time operation. The third block is the Inverter block, which accurately represents the inverter's operation by taking the PWM commands from the controller and the DC link voltage as inputs. It stimulates the switching behavior of the power electronic devices, outputting the phase voltages supplied to the motor. The fourth block is the PMSM block, previously developed in Chapter 4. This block simulates the motor's electrical and mechanical responses to the inverter inputs. It takes the phase voltages from the inverter model and produces outputs such as motor torque and speed. The motor model also provides feedback signals to the controller, closing the simulation loop and allowing for adjustments based on the system's dynamic behavior.

5.6 EV Powertrain Hybrid Model Validation

The validation of the hybrid model is a crucial step to ensure that all its components function perfectly and seamlessly. This step incorporates using new reference scenarios of speed and torque that were not used during the development phase. These scenarios are specifically designed to assess the ability of the data-driven components to accurately predict new data and supply inputs to the model-based components (motor and inverter) and effectively utilize the feedback from the motor model. To facilitate this process, modifications were made to the sample times of each block to ensure unified coordination between the various models. The specifics of validation tests are detailed in Table 9.

Table 9. Validation test cases.

Test	Ref. Speed (p.u.)	Ref. Torque (p.u.)
X	0.16	1.00
Y	0.85	0.21

Figures 26–27 (a, b, c) compare the hybrid model and the physical system performance under the applied test conditions.

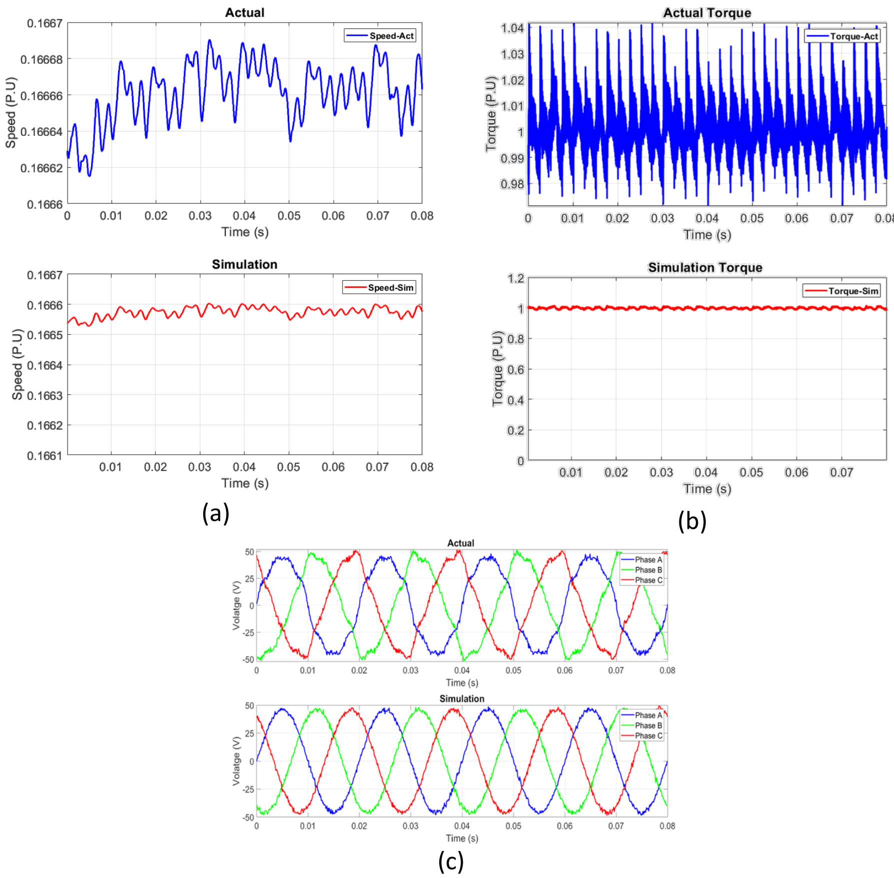


Figure 26. Case (X) Actual and virtual speed (a), torque (b), and input voltage (c).

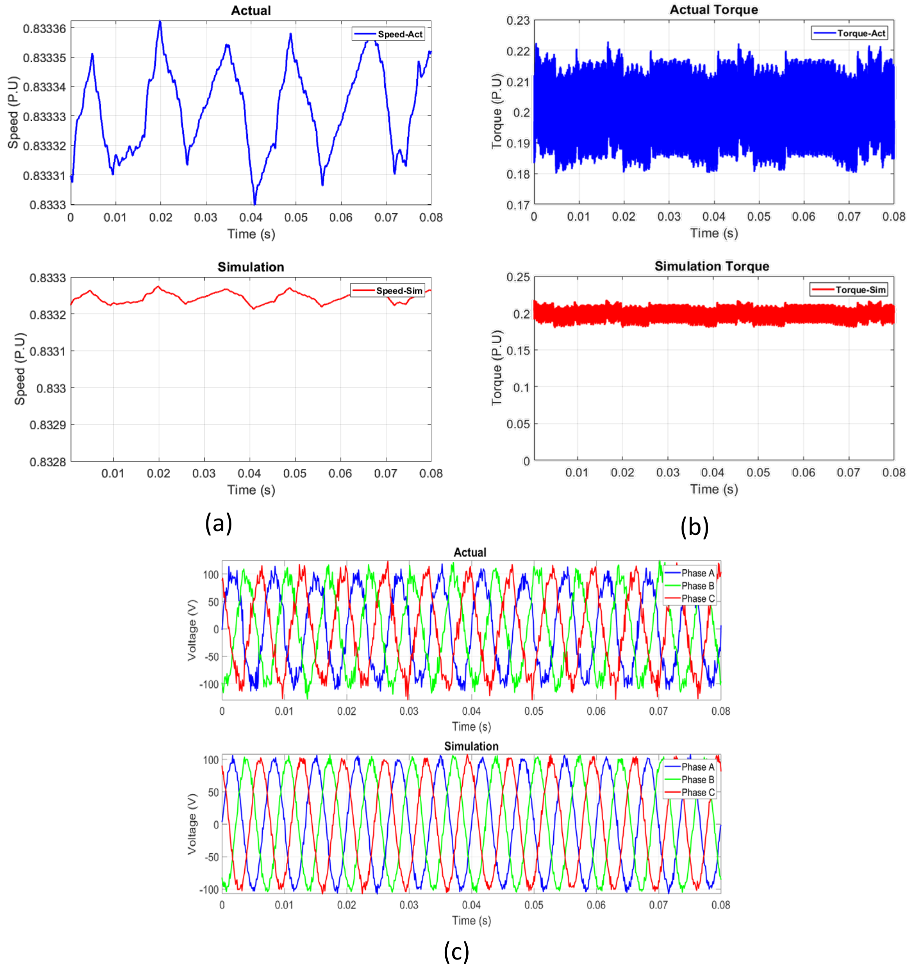


Figure 27. Case (Y) actual and virtual speed (a), torque (b), and input voltage (c).

The comparison between the virtual and actual results, as shown in Figure 25 a, highlights that the hybrid model generates less ripple in the steady-state speed compared to the observed real-world data. This discrepancy is primarily due to the presence of noise in measurement devices. Despite this, the model's ability to accurately match the average speed value with the real data demonstrates its high fidelity and successful emulation of the system's dynamics. For torque simulation, as depicted in Figure 26 (b), the hybrid model achieved a smoother and more consistent output than the real torque measurements. This difference likely stems from varying real-world conditions that are challenging to replicate in a simulated environment. In Case Y, representing the CP region at high speeds, Figure 27 (a) illustrates a strong agreement between simulated and actual speed, with the simulation exhibiting greater stability. This emphasizes the model's ability to capture system performance accurately under high-speed conditions. Furthermore, Figure 27 (b) reveals that both simulated and actual torque signals display similar ripple patterns, demonstrating the model's capability to replicate torque fluctuations observed in real-world operations. The detailed comparisons, with an average MSE of 3%, underscore the hybrid model's robustness in replicating real-world conditions. The results in Figures 26 (c) and 27 (c) provide the strongest evidence of the model's validity in accurately reproducing the drive system's performance.

5.7 Hybrid Model Deployment to Digital Twin Framework

To evaluate the suitability of the hybrid model for DT application, it was deployed onto a real-time target machine (Speed goat Baseline). This deployment allowed the model to execute in real-time, with a host computer providing dynamic control commands and monitoring the system's response. This setup enabled a comprehensive assessment of the model's computational speed and efficiency under realistic operating conditions. The hybrid model was compiled and deployed onto the target machine using Simulink Real-Time, ensuring precise execution on dedicated hardware. A host model, developed in Simulink, was the control interface to interact with the target model, transmitting real-time commands and receiving continuous feedback.

The communication between the host and target models utilized the UDP protocol to facilitate efficient and high-speed data exchange. This choice ensured rapid command transmission with minimal latency, meeting the demands of real-time interaction. The host model continuously transmitted speed and torque commands to the target model, which processed these inputs and provided real-time feedback. This dynamic interaction enabled rigorous testing of the model's computational performance.

To simulate real-world scenarios, the actual EV (ISEAUTO) drive system was instrumented with sensors to measure voltage, current, speed, and torque. The vehicle was operated on a designated track within the university campus, chosen to represent diverse driving conditions. The test began with the vehicle at a standstill, gradually increasing velocity until reaching a near-constant speed. The track's altitude gradually rose from 22.4 m to 23.2 m above sea level, introducing a slight increase in motor load torque. At the end of the test, the EV decelerated to a stop at a designated parking spot. Data from the experiment was collected using a DAS and stored as MATLAB files, capturing realistic driving behaviors such as acceleration, deceleration, and steady-state operation. This data was then preprocessed following the procedures described in Section IV.B to ensure high quality and consistency for subsequent analysis. The torque and speed data from the experiment were repurposed as command inputs for the host model, recreating real-world driving conditions for testing the hybrid model. In this configuration, the host model transmitted speed and torque commands in real-time via UDP to the target machine, where the hybrid model processed them and generated corresponding outputs, including phase voltage, feedback speed, and torque. This setup demonstrated the hybrid model's capability to handle real-time data processing and accurately emulate the behavior of the EV drive system under realistic conditions. Figure 28 illustrates the DT framework structure.

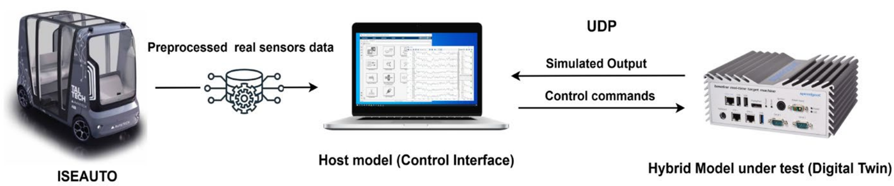
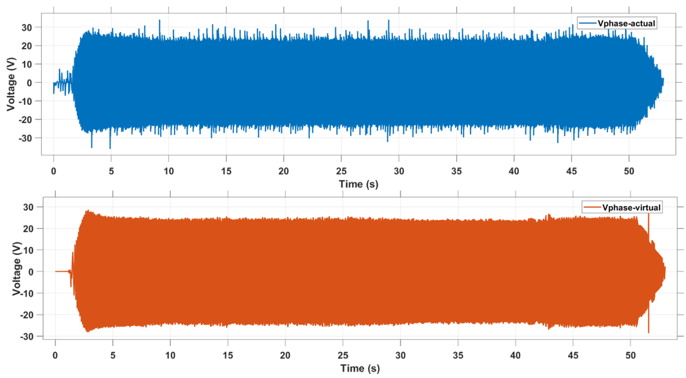
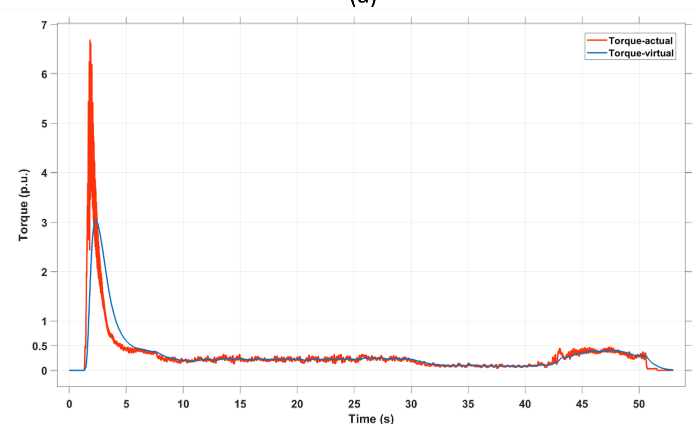


Figure 28. DT testing framework.

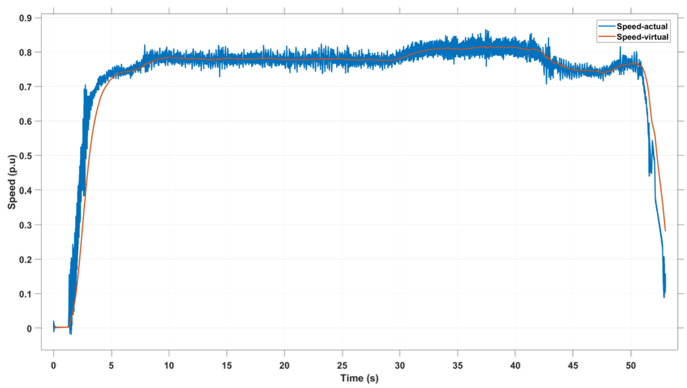
The outputs from the target machine were continuously monitored and compared with the actual data obtained from the EV powertrain. This comparison precisely evaluated the hybrid model's capability to replicate real-world behavior. Figure 29 (a, b, c) presents a detailed performance comparison between the real drive system and the hybrid model, highlighting their alignment and validating the model's accuracy in emulating the dynamics of the EV drive system.



(a)



(b)



(c)

Figure 29. Performance comparison between the real EV powertrain and developed hybrid model, as follows voltage (a), torque (b), and speed (c).

The results presented in Figure 29 comprehensively evaluate the hybrid model's performance. In Figure 29 (a), a noticeable discrepancy is observed between the voltage waveforms of the real system and the hybrid model. The real voltage signal exhibits higher noise and fluctuations, attributed to sensor variability, whereas the hybrid model outputs a smoother waveform, reflecting the refined and pre-processed data used during training. Despite these differences, the RMS values of the real and predicted voltages show close agreement, with an error margin of less than 4%. This indicates that the hybrid model effectively captures the average voltage behavior while filtering out minor fluctuations. In Figure 29 (b), the torque signal highlights a significant mismatch during the initial torque phase when the vehicle transitions from a standstill. The hybrid model struggles to fully replicate the transient behavior, likely due to limitations in the NARX model, which was trained on preprocessed test bench data that did not entirely capture the initial torque dynamics of real-world conditions. Additionally, the smoother transitions in the hybrid model's prediction may be influenced by sensor constraints in the test bench data. However, the model demonstrates high accuracy under steady-state conditions, closely mirroring the real system's torque behavior, confirming its reliability in stable operational scenarios. Finally, Figure 29 (c) illustrates a strong alignment between the simulated and real speed outputs, with the hybrid model exhibiting a notably smoother response and reduced ripple. This outcome is expected, as the simulation environment lacks the noise and disturbances inherent in physical measurements.

Overall, the results demonstrate that the hybrid model provides a robust representation of the EV powertrain's behavior, particularly under steady-state conditions, making it well-suited for applications such as condition monitoring, performance evaluation, and predictive maintenance. The model's ability to accurately replicate voltage, torque, and speed under stable operating conditions ensures its effectiveness in real-time monitoring systems where consistency and efficiency are critical.

However, the model's limitations in capturing transient behaviors, such as the initial torque response during start-stop scenarios, indicate areas for improvement. These shortcomings suggest that while the model is reliable for steady-state analysis, further refinements are needed for applications requiring precise transient performance, such as fault detection, dynamic load analysis, and control strategy optimization. Enhancing data collection processes and incorporating more diverse training methodologies may improve the model's ability to address transient dynamics, broadening its applicability in dynamic, real-world environments.

5.8 Real-time Performance Evaluation

The dynamic interaction between the host and target systems provided valuable insights into the hybrid model's computational performance, stability, and responsiveness under realistic operating conditions. The model's ability to respond to changing inputs was evaluated by continuously transmitting speed and torque commands from the host, enabling direct assessment of response time, stability, and computational efficiency. The hybrid model achieved an average cycle time of 8 ms per iteration, consistently meeting the 10 ms threshold required for real-time EV control. This performance remained stable across various test conditions, demonstrating the model's robustness in handling rapid input changes without any noticeable lag. Additionally, the target machine exhibited efficient resource utilization, with an average CPU load of 60% and memory usage of 300 MB, confirming stable system operation throughout the testing process.

The communication between the host and target systems via UDP further validated the model's capability to respond to commands and update outputs with minimal latency. This seamless interaction underscores the hybrid model's suitability for DT applications, where real-time responsiveness and computational efficiency are essential.

5.9 Chapter Summary

This chapter delved into the modeling, validation, and deployment to DT framework of the EV powertrain, focusing on achieving real-time capabilities and high-fidelity representation. Due to inherent uncertainties in the drive system, a data-driven approach was adopted to model its behavior. Initially, the drive system was represented through an elementary model, where the entire system was treated as a single block. This model utilized DANN to provide preliminary insights into system behavior. While effective for gaining a general understanding of system dynamics, the elementary model's oversimplification limited its ability to capture the nuanced interactions between the drive system's core components, the inverter, and the controller – rendering it unsuitable for complex scenarios. To address these limitations, an advanced model was developed, dividing the drive system into its two fundamental components: the inverter (frequency converter) and the controller. The controller, with its nonlinear behavior and parameter uncertainties, was modeled using an NARX. This approach effectively captured the time-dependent and nonlinear dynamics of the controller. On the other hand, the inverter, characterized by its well-defined electrical properties, was modeled using a physics-based approach. By separating these components, the advanced model significantly enhanced the accuracy and granularity of the drive system's representation. The culmination of this work was the development of a hybrid model, which was developed by integrating a detailed physics-based motor model with the data-driven model of the drive system. offering a comprehensive simulation of the EV powertrain. Validation against empirical test bench data demonstrated the model's high fidelity, achieving 97% accuracy in replicating real-world performance. The hybrid model was subsequently deployed on a real-time target machine to evaluate its suitability for DT applications. During this evaluation, the model exhibited stability, responsiveness, and efficient resource utilization, consistently meeting real-time control requirements with an average cycle time of 8 ms and a CPU load of 60%. However, during real-time testing, the model showed limitations in accurately capturing transient behaviors during starting conditions. This shortcoming is likely attributed to the challenges of replicating real-world noise and rapid fluctuations that were not fully represented in the preprocessed training data. In contrast, the model demonstrated excellent performance in steady-state conditions, accurately replicating system dynamics and confirming its reliability for stable operation. Figure 30 shows the procedures for developing the powertrain hybrid model DT.

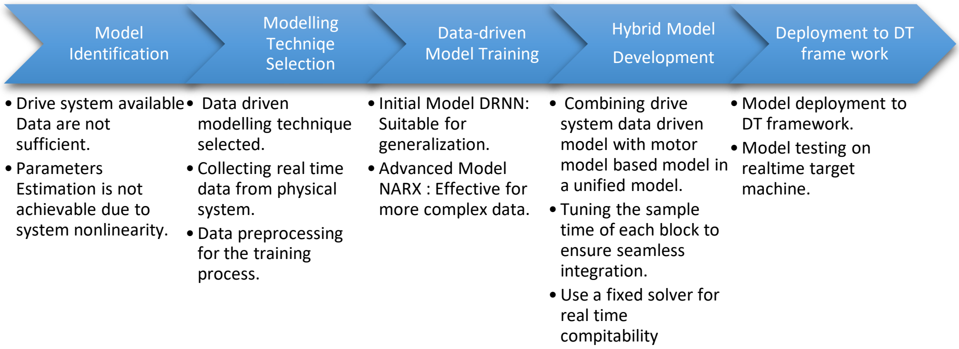


Figure 30. Development framework of the EV powertrain DT model.

6 Conclusions and Future Work

6.1 Conclusions

This thesis proposed an advanced modeling framework for the DT of an electric EV propulsion drive system, providing a comprehensive guide for researchers and developers to better understand the nature of DTs and their operational mechanisms. Beginning with an extensive overview of DTs – including their types, applications, historical development, and recent advancements – the framework laid the foundation for a detailed examination of the system under study: the EV propulsion drive system. Focusing on the motor drive system, which comprises the motor, inverter, and controller, the research identified it as the most critical part to model due to its complexity and the intricate dynamics involved. Each component presented unique challenges in representation, necessitating a division into main components to tailor modeling approaches effectively.

The choice of modeling techniques depended on several key factors, including the availability of system parameters, the complexity of each component, the level of detail required, computational efficiency, and the intended application of the model. For instance, physics-based (model-based) approaches were utilized for components like the motor, where existing knowledge of system dynamics was well-established. In contrast, data-driven modeling techniques were applied to components like the inverter and controller to enhance overall model accuracy and address limitations in the physics-based approach, especially when dealing with partially unknown parameters. Reduced-order modeling techniques were implemented to improve computational speed without significantly sacrificing accuracy, which is crucial for real-time applications. Robust model identification technique was incorporated into the motor model to handle partially unknown physical parameters, ensuring precise representation and managing uncertainties effectively.

The outcomes of this research include the development and validation of high-precision digital models that accurately replicate physical system performance while achieving high computational speed. The models demonstrated the ability to predict system behavior accurately under various operating conditions, confirmed through comprehensive validation against real-world data. Optimization of model structures facilitated efficient communication and data exchange with physical systems, enhancing integration and practical deployment.

Furthermore, the developed digital models were successfully deployed into DT framework in both simulated and real-world environments, evaluating their performance, integration, and tangible benefits in actual operation. This work advances the understanding and application of DT technology in EV propulsion systems and provides a versatile framework for future research and development. By addressing the challenges posed by system uncertainties and partially unknown parameters, and by leveraging both model-based and data-driven approaches, this thesis lays a solid foundation for further innovations in sustainable transportation technologies, promoting the integration of DTs into a broader range of applications within the automotive industry and beyond.

6.2 Future Directions

Future work will focus on improving models' accuracy, expanding the scope of DT functions, and enhancing real-time adaptability. One key area for improvement is the refinement of transient dynamics modeling, ensuring that the DT can capture rapid load changes, initial torque responses, and short-duration fluctuations more accurately. This will require incorporating advanced training techniques and diverse datasets to improve the model's generalization to high-frequency disturbances.

Another crucial direction is the expansion of fault detection and predictive maintenance capabilities by integrating real-time diagnostic tools and adaptive monitoring algorithms. By leveraging dynamic load analysis, future research can further optimize control strategies, enabling more effective identification of potential system failures and preventive intervention strategies. The implementation of machine learning-based anomaly detection will allow DTs to continuously learn from operational data, improving reliability and efficiency.

Additionally, the evolution of SDEV presents new opportunities for DT modularity. Future research would explore modular DT architectures that enable flexible, scalable, and upgradable vehicle systems, allowing for seamless integration of software updates, adaptive control algorithms, and cloud-based computation. This will contribute to the development of next-generation smart vehicle platforms, where DTs serve as intelligent decision-making tools for optimized energy management, performance adaptation, and fault tolerance in electric propulsion systems.

List of Figures

List of Tables

References

[1] "Global EV growth forecast in 2024, but challenges remain." Accessed: Dec. 20, 2024. [Online]. Available: https://ev-volumes.com/news/ev/global-ev-growth-forecast-in-2024-but-challenges-remain/?utm_source=chatgpt.com

[2] N. Murali, V. P. Mini, and S. Ushakumari, "Electric Vehicle Market Analysis and Trends," *INDICON 2022 - 2022 IEEE 19th India Council International Conference*, 2022, doi: 10.1109/INDICON56171.2022.10040056.

[3] V. Rjabtšikov, A. Rassõlkin, K. Kudelina, A. Kallaste, and T. Vaimann, "Review of Electric Vehicle Testing Procedures for Digital Twin Development: A Comprehensive Analysis," *Energies 2023, Vol. 16, Page 6952*, vol. 16, no. 19, p. 6952, Oct. 2023, doi: 10.3390/EN16196952.

[4] Z. Liu, W. Zhang, and F. Zhao, "Impact, Challenges and Prospect of Software-Defined Vehicles," *Automotive Innovation*, vol. 5, no. 2, pp. 180–194, Apr. 2022, doi: 10.1007/S42154-022-00179-Z/FIGURES/11.

[5] D. Goldsman, R. E. Nance, and J. R. Wilson, "A brief history of simulation," *Proceedings - Winter Simulation Conference*, pp. 310–313, 2009, doi: 10.1109/WSC.2009.5429341.

[6] A. Wimo, R. Handels, and L. Jönsson, "The art of simulation," *Lancet Healthy Longev*, vol. 1, no. 1, pp. e2–e3, Oct. 2020, doi: 10.1016/S2666-7568(20)30006-4.

[7] S. K. Lahiri, "Offline Simulation," *Multivariable Predictive Control*, pp. 167–182, Aug. 2017, doi: 10.1002/9781119243434.CH11.

[8] S. Nabi, M. Balike, J. Allen, and K. Rzemien, "An overview of hardware-in-The-loop testing systems at Visteon," *SAE Technical Papers*, 2004, doi: 10.4271/2004-01-1240.

[9] M. Ibrahim, A. Rassõlkin, T. Vaimann, and A. Kallaste, "Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System," *Sustainability 2022, Vol. 14, Page 601*, vol. 14, no. 2, p. 601, Jan. 2022, doi: 10.3390/SU14020601.

[10] S. Nabi, M. Balike, J. Allen, and K. Rzemien, "An overview of hardware-in-The-loop testing systems at Visteon," *SAE Technical Papers*, 2004, doi: 10.4271/2004-01-1240.

[11] Z. Chen and L. Huang, "Digital twins for information-sharing in remanufacturing supply chain: A review," *Energy*, vol. 220, Apr. 2021, doi: 10.1016/J.ENERGY.2020.119712.

[12] Y. Zheng, S. Yang, and H. Cheng, "An application framework of digital twin and its case study," *Journal of Ambient Intelligence and Humanized Computing 2018 10:3*, vol. 10, no. 3, pp. 1141–1153, Jun. 2018, doi: 10.1007/S12652-018-0911-3.

[13] A. M. Madni, C. C. Madni, and S. D. Lucero, "Leveraging Digital Twin Technology in Model-Based Systems Engineering," *Systems 2019, Vol. 7, Page 7*, vol. 7, no. 1, p. 7, Jan. 2019, doi: 10.3390/SYSTEMS7010007.

[14] A. Rassõlkin *et al.*, "Implementation of Digital Twins for electrical energy conversion systems in selected case studies," *Proceedings of the Estonian Academy of Sciences*, vol. 70, no. 1, 2021.

[15] A. Rassolkin, T. Vaimann, A. Kallaste, and V. Kuts, “Digital twin for propulsion drive of autonomous electric vehicle,” in *2019 IEEE 60th Annual International Scientific Conference on Power and Electrical Engineering of Riga Technical University, RTUCON 2019 - Proceedings*, IEEE, Oct. 2019, pp. 1–4. doi: 10.1109/RTUCON48111.2019.8982326.

[16] A. Rassõlkin *et al.*, “Implementation of digital twins for electrical energy conversion systems in selected case studies,” *Proceedings of the Estonian Academy of Sciences*, vol. 70, no. 1, pp. 19–39, 2021, doi: 10.3176/PROC.2021.1.03.

[17] F. Biesinger and M. Weyrich, “The Facets of Digital Twins in Production and the Automotive Industry,” *2019 23rd International Conference on Mechatronics Technology, ICMT 2019*, Oct. 2019, doi: 10.1109/ICMECT.2019.8932101.

[18] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, “Characterising the Digital Twin: A systematic literature review,” *CIRP J Manuf Sci Technol*, vol. 29, pp. 36–52, May 2020, doi: 10.1016/J.CIRPJ.2020.02.002.

[19] M. G. Juarez, V. J. Botti, and A. S. Giret, “Digital twins: Review and challenges,” *J Comput Inf Sci Eng*, vol. 21, no. 3, Jun. 2021, doi: 10.1115/1.4050244/1100428.

[20] A. Fuller, Z. Fan, C. Day, and C. Barlow, “Digital Twin: Enabling Technologies, Challenges and Open Research,” *IEEE Access*, vol. 8, pp. 108952–108971, 2020, doi: 10.1109/ACCESS.2020.2998358.

[21] Y. Lu, C. Liu, K. I. K. Wang, H. Huang, and X. Xu, “Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues,” *Robot Comput Integr Manuf*, vol. 61, Feb. 2020, doi: 10.1016/J.RCIM.2019.101837.

[22] Bottani E, Cammardella A, Murino T, and Vespoli S, “From the Cyber-Physical System to the Digital Twin: the process development for behaviour modelling of a Cyber Guided Vehicle in M2M logic”.

[23] E. H. Glaessgen and D. S. Stargel, “The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles”.

[24] W. M. P. van der Aalst, O. Hinz, and C. Weinhardt, “Resilient Digital Twins: Organizations Need to Prepare for the Unexpected,” *Business and Information Systems Engineering*, vol. 63, no. 6, pp. 615–619, Dec. 2021, doi: 10.1007/S12599-021-00721-Z/FIGURES/2.

[25] A. M. Madni, C. C. Madni, and S. D. Lucero, “Leveraging Digital Twin Technology in Model-Based Systems Engineering,” *Systems 2019, Vol. 7, Page 7*, vol. 7, no. 1, p. 7, Jan. 2019, doi: 10.3390/SYSTEMS7010007.

[26] A. M. Madni, C. C. Madni, and S. D. Lucero, “Leveraging Digital Twin Technology in Model-Based Systems Engineering,” *Systems 2019, Vol. 7, Page 7*, vol. 7, no. 1, p. 7, Jan. 2019, doi: 10.3390/SYSTEMS7010007.

[27] G. Bachelor, E. Brusa, D. Ferretto, and A. Mitschke, “Model-Based Design of Complex Aeronautical Systems through Digital Twin and Thread Concepts,” *IEEE Syst J*, vol. 14, no. 2, pp. 1568–1579, Jun. 2020, doi: 10.1109/JSYST.2019.2925627.

[28] M. C. Magnanini and T. A. M. Tolio, “A model-based Digital Twin to support responsive manufacturing systems,” *CIRP Annals*, vol. 70, no. 1, pp. 353–356, Jan. 2021, doi: 10.1016/J.CIRP.2021.04.043.

[29] P. Zheng and A. S. Sivabalan, "A generic tri-model-based approach for product-level digital twin development in a smart manufacturing environment," *Robot Comput Integr Manuf*, vol. 64, p. 101958, Aug. 2020, doi: 10.1016/J.RCIM.2020.101958.

[30] Y. Li and X. Shen, "A Novel Wind Speed-Sensing Methodology for Wind Turbines Based on Digital Twin Technology," *IEEE Trans Instrum Meas*, vol. 71, 2022, doi: 10.1109/TIM.2021.3139698.

[31] R. Woitsch, A. Sumereder, and D. Falcioni, "Model-based data integration along the product & service life cycle supported by digital twinning," *Comput Ind*, vol. 140, p. 103648, Sep. 2022, doi: 10.1016/J.COMPIND.2022.103648.

[32] R. He, G. Chen, C. Dong, S. Sun, and X. Shen, "Data-driven digital twin technology for optimized control in process systems," *ISA Trans*, vol. 95, pp. 221–234, Dec. 2019, doi: 10.1016/J.ISATRA.2019.05.011.

[33] Z. Wang *et al.*, "Mobility Digital Twin with Connected Vehicles and Cloud Computing Digital Twin for Connected Vehicles View project CACC Enabled EAD Application View project Mobility Digital Twin : Concept, Architecture, Case Study, and Future Challenges," 2022, doi: 10.36227/techrxiv.16828759.v1.

[34] C. Gao, H. Park, and A. Easwaran, "An anomaly detection framework for digital twin driven cyber-physical systems," *ICCPS 2021 - Proceedings of the 2021 ACM/IEEE 12th International Conference on Cyber-Physical Systems (with CPS-IoT Week 2021)*, vol. 11, no. 21, pp. 44–54, May 2021, doi: 10.1145/3450267.3450533.

[35] A. Coraddu, L. Oneto, F. Baldi, F. Cipollini, M. Atlar, and S. Savio, "Data-driven ship digital twin for estimating the speed loss caused by the marine fouling," *Ocean Engineering*, vol. 186, p. 106063, Aug. 2019, doi: 10.1016/J.OCEANENG.2019.05.045.

[36] K. Mykoniatis and G. A. Harris, "A digital twin emulator of a modular production system using a data-driven hybrid modeling and simulation approach," *J Intell Manuf*, vol. 32, no. 7, pp. 1899–1911, Oct. 2021, doi: 10.1007/S10845-020-01724-5/TABLES/3.

[37] C. Blume, S. Blume, S. Thiede, and C. Herrmann, "Data-Driven Digital Twins for Technical Building Services Operation in Factories: A Cooling Tower Case Study," *Journal of Manufacturing and Materials Processing 2020, Vol. 4, Page 97*, vol. 4, no. 4, p. 97, Sep. 2020, doi: 10.3390/JMMP4040097.

[38] W. Kim, S. Kim, J. Jeong, H. Kim, H. Lee, and B. D. Youn, "Digital twin approach for on-load tap changers using data-driven dynamic model updating and optimization-based operating condition estimation," *Mech Syst Signal Process*, vol. 181, p. 109471, Dec. 2022, doi: 10.1016/J.YMSSP.2022.109471.

[39] L. Campanile, M. S. De Biase, R. De Fazio, M. Di Giovanni, F. Marulli, and L. Verde, "Merging Model-Based and Data-Driven Approaches for Resilient Systems Digital Twins Design," *Proceedings of the 2023 IEEE International Conference on Cyber Security and Resilience, CSR 2023*, pp. 301–306, 2023, doi: 10.1109/CSR57506.2023.10224945.

[40] M. G. Kapteyn, D. J. Knezevic, D. B. P. Huynh, M. Tran, and K. E. Willcox, “Data-driven physics-based digital twins via a library of component-based reduced-order models,” *Int J Numer Methods Eng*, vol. 123, no. 13, pp. 2986–3003, Jul. 2022, doi: 10.1002/NME.6423.

[41] A. Wunderlich, K. Booth, and E. Santi, “Hybrid Analytical and Data-Driven Modeling Techniques for Digital Twin Applications,” *2021 IEEE Electric Ship Technologies Symposium, ESTS 2021*, Aug. 2021, doi: 10.1109/ESTS49166.2021.9512364.

[42] M. G. Kapteyn, D. J. Knezevic, D. B. P. Huynh, M. Tran, and K. E. Willcox, “Data-driven physics-based digital twins via a library of component-based reduced-order models,” *Int J Numer Methods Eng*, vol. 123, no. 13, pp. 2986–3003, Jul. 2022, doi: 10.1002/NME.6423.

[43] H. Ejersbo, K. Lausdahl, M. Frasheri, and L. Esterle, “Dynamic Runtime Integration of New Models in Digital Twins,” *ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems*, vol. 2023-May, pp. 44–55, 2023, doi: 10.1109/SEAMS59076.2023.00016.

[44] C. L. Cai, X. G. Wang, Y. W. Bai, Y. C. Xia, and K. Liu, “Key technologies of EV motor drive system design,” *Procedia Eng*, vol. 16, pp. 356–362, 2011, doi: 10.1016/j.proeng.2011.08.1095.

[45] B. Wu, W. D. Widanage, S. Yang, and X. Liu, “Battery digital twins: Perspectives on the fusion of models, data and artificial intelligence for smart battery management systems,” *Energy and AI*, vol. 1, p. 100016, Aug. 2020, doi: 10.1016/J.EGYAI.2020.100016.

[46] W. Wang, J. Wang, J. Tian, J. Lu, and R. Xiong, “Application of Digital Twin in Smart Battery Management Systems,” *Chinese Journal of Mechanical Engineering 2021 34:1*, vol. 34, no. 1, pp. 1–19, Jun. 2021, doi: 10.1186/S10033-021-00577-0.

[47] N. A. Elsonbaty, M. A. Enany, and M. I. Hassanin, “An Efficient Vector Control Policy for EV-Hybrid Excited Permanent-Magnet Synchronous Motor,” *World Electric Vehicle Journal 2020, Vol. 11, Page 42*, vol. 11, no. 2, p. 42, May 2020, doi: 10.3390/WEVJ11020042.

[48] P. Bhatt, H. Mehar, and M. Sahajwani, “Electrical Motors for Electric Vehicle – A Comparative Study,” *SSRN Electronic Journal*, Apr. 2019, doi: 10.2139/SSRN.3364887.

[49] M. Ibrahim, A. Rassõlkin, T. Vaimann, and A. Kallaste, “Propulsion drives and control algorithms of electrical vehicles,” *Handb. Power Electron. Auton. Electr. Veh.*, pp. 139–159, Jan. 2024, doi: 10.1016/B978-0-323-99545-0.00009-9.

[50] J. Xia, L. Wang, and Y. Zhou, “Design and Development of Digital Twin System for Intelligent Motor,” *Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020*, pp. 991–996, Aug. 2020, doi: 10.1109/AEECA49918.2020.9213679.

[51] S. Venkatesan, K. Manickavasagam, N. Tengenkai, and N. Vijayalakshmi, “Health monitoring and prognosis of electric vehicle motor using intelligent-digital twin,” *IET Electr Power Appl*, vol. 13, no. 9, pp. 1328–1335, Sep. 2019, doi: 10.1049/IET-EPA.2018.5732.

[52] V. Rjabtsikov *et al.*, "Digital Twin Service Unit for AC Motor Stator Inter-Turn Short Circuit Fault Detection," *2021 28th International Workshop on Electric Drives: Improving Reliability of Electric Drives, IWED 2021 - Proceedings*, Jan. 2021, doi: 10.1109/IWED52055.2021.9376328.

[53] F. Toso, R. Torchio, A. Favato, P. G. Carlet, S. Bolognani, and P. Alotto, "Digital twins as electric motor soft-sensors in the automotive industry," in *2021 IEEE International Workshop on Metrology for Automotive, MetroAutomotive 2021 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Jul. 2021, pp. 13–18. doi: 10.1109/MetroAutomotive50197.2021.9502885.

[54] J. F. D. Santos *et al.*, "Digital Twin-Based Monitoring System of Induction Motors Using IoT Sensors and Thermo-Magnetic Finite Element Analysis," *IEEE Access*, vol. 11, pp. 1682–1693, 2023, doi: 10.1109/ACCESS.2022.3232063.

[55] A. Rassolkin, V. Rjabtsikov, T. Vaimann, A. Kallaste, V. Kuts, and A. Partyshev, "Digital Twin of an Electrical Motor Based on Empirical Performance Model," in *2020 11th International Conference on Electrical Power Drive Systems, ICEPDS 2020 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020. doi: 10.1109/ICEPDS47235.2020.9249366.

[56] H. Brandtstaedter, L. Hübner, A. Jungiewicz, C. Ludwig, E. Tsouchnika, and U. Wever, "DIGITAL TWINS FOR LARGE ELECTRIC DRIVE TRAINS Copyright Material PCIC Europe Paper No. PCIC Europe EUR18_04 Germany Germany Germany," 2018.

[57] J. Xia, L. Wang, and Y. Zhou, "Design and Development of Digital Twin System for Intelligent Motor," *Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020*, pp. 991–996, Aug. 2020, doi: 10.1109/AEECA49918.2020.9213679.

[58] S. Venkatesan, K. Manickavasagam, N. Tengenkai, and N. Vijayalakshmi, "Health monitoring and prognosis of electric vehicle motor using intelligent-digital twin," *IET Electr Power Appl*, vol. 13, no. 9, pp. 1328–1335, Sep. 2019, doi: 10.1049/IET-EPA.2018.5732.

[59] M. Ibrahim, A. Rassõlkin, T. Vaimann, and A. Kallaste, "Propulsion drives and control algorithms of electrical vehicles," *Handbook of Power Electronics in Autonomous and Electric Vehicles*, pp. 139–159, Jan. 2024, doi: 10.1016/B978-0-323-99545-0.00009-9.

[60] M. Milton, C. O. De La, H. L. Ginn, and A. Benigni, "Controller-Embeddable Probabilistic Real-Time Digital Twins for Power Electronic Converter Diagnostics," *IEEE Trans Power Electron*, vol. 35, no. 9, pp. 9852–9866, Sep. 2020, doi: 10.1109/TPEL.2020.2971775.

[61] A. Wunderlich and E. Santi, "Digital twin models of power electronic converters using dynamic neural networks," in *Conference Proceedings - IEEE Applied Power Electronics Conference and Exposition - APEC*, Institute of Electrical and Electronics Engineers Inc., Jun. 2021, pp. 2369–2376. doi: 10.1109/APEC42165.2021.9487201.

[62] Y. Liu, G. Chen, Y. Liu, L. Mo, and X. Qing, “Condition Monitoring of Power Electronics Converters Based on Digital Twin,” *2021 IEEE 3rd International Conference on Circuits and Systems, ICCS 2021*, pp. 190–195, 2021, doi: 10.1109/ICCS52645.2021.9697303.

[63] Q. Wu, W. Wang, Q. Wang, L. Xiao, and B. Hu, “Digital Twin Approach for Degradation Parameters Identification of a Single-Phase DC-AC Inverter,” *Conference Proceedings - IEEE Applied Power Electronics Conference and Exposition - APEC*, pp. 1725–1730, 2022, doi: 10.1109/APEC43599.2022.9773462.

[64] H. Shi, L. Xiao, Q. Wu, and W. Wang, “Digital Twin Approach for IGBT Parameters Identification of a Three-Phase DC-AC Inverter,” *2022 IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific)*, pp. 1–4, Oct. 2022, doi: 10.1109/ITECASIA-PACIFIC56316.2022.9942105.

[65] X. Liu, M. Hofmann, F. Streit, and M. Maerz, “Digital Twin for Intelligent and SiC-based Drive Systems,” *2021 11th International Electric Drives Production Conference, EDPC 2021 - Proceedings*, 2021, doi: 10.1109/EDPC53547.2021.9684201.

[66] B. Gao *et al.*, “Topology optimization and the evolution trends of two-speed transmission of EVs,” *Renewable and Sustainable Energy Reviews*, vol. 161, p. 112390, Jun. 2022, doi: 10.1016/J.RSER.2022.112390.

[67] E. Sekhri, M. Ibrahim, R. G. Zequera, and A. Rassõlkin, “Electric and Hybrid Vehicles,” *Smart Electric and Hybrid Vehicles*, pp. 71–126, Feb. 2025, doi: 10.1002/9781394225040.CH3.

[68] J. Hu, N. Hu, P. Luo, and Y. Yang, “Fault Diagnosis of Gearbox Based on Digital Twin Concept Model,” *4th International Conference on Intelligent Robotics and Control Engineering, IRCE 2021*, pp. 30–34, 2021, doi: 10.1109/IRCE53649.2021.9570997.

[69] “2017 Mitsubishi i-MiEV Technical and Mechanical Features.” Accessed: Dec. 03, 2024. [Online]. Available: https://media.mitsubishicars.com/en-US/releases/2017-mitsubishi-i-miev-technical-and-mechanical-features

[70] A. Rassõlkin, R. Sell, and M. Leier, “Development case study of the first estonian self-driving car, iseauto,” *Electrical, Control and Communication Engineering*, vol. 14, no. 1, pp. 81–88, Jul. 2018, doi: 10.2478/ECCE-2018-0009.

[71] S. Mirjalili, “The Ant Lion Optimizer,” *Advances in Engineering Software*, vol. 83, pp. 80–98, May 2015, doi: 10.1016/J.ADVENGSOFT.2015.01.010.

Acknowledgement

First and foremost, I am profoundly grateful to my family for their unwavering support and encouragement throughout this journey. To my beloved father, Ibrahim, who recently passed away – this work is dedicated to you. Your values, wisdom, and unwavering belief in me have been a guiding light, even in your absence. I have always wished you could be here to celebrate this milestone with us, but I take comfort in knowing that your soul is with me every step of the way. I hope you are in a place of peace and rest, smiling down on me.

To my dearest mother, Aziza, your endless love, prayers, and encouragement have been my anchor throughout this journey. Your sacrifices and unwavering belief in me are what made this achievement possible, and I am forever grateful.

To my wife, Sara, thank you for your patience, love, and understanding during this demanding journey. Your unwavering support and belief in me have been my strength through every challenge, and I could not have done this without you by my side.

I want to express my deepest gratitude to my supervisor, Professor Anton Rassõlkin, for his invaluable guidance, support, and mentorship throughout my PhD. Your insights and encouragement have been instrumental in shaping this work and helping me grow as a researcher.

Finally, thanks to my colleagues at TalTech Mechatronics and Autonomous systems Research Group for their endless support during my journey.

Thank you to everyone who has been part of this journey in any way – your contributions, no matter how big or small, mean the world to me.

Abstract

Advanced Modelling Frameworks for The Digital Twin of an Autonomous Electric Vehicle Propulsion Drive System

The increasing demand for efficient, reliable, and sustainable electric vehicles (EVs) has driven the development of innovative technologies, with Digital Twins (DTs) emerging as a transformative tool. A DT is a real-time virtual replica of a physical system that enables continuous monitoring, analysis, and optimization throughout the system's lifecycle. In the context of EV propulsion systems, DTs provide unparalleled opportunities to enhance performance, predictive maintenance, and system adaptability, marking a significant shift from traditional modeling and simulation methods.

This thesis presents a comprehensive framework for developing a DT of an EV propulsion drive system, addressing the challenges posed by system complexities, uncertainties, and partially unknown parameters. The work begins with a systematic overview of the evolution of DTs and their applications in EVs, highlighting widely used platforms and the specific applications of DT technology in EV propulsion systems. It then introduces the system under study, focusing on the propulsion system of ISEAUTO, a second-life repurposed EV. The propulsion system's architecture and unique characteristics are explored in detail to provide the foundational context for model development. The thesis delves into various modeling techniques, detailing the selection criteria for each and their suitability for representing different system components. A hybrid modeling approach is proposed, combining physics-based and data-driven techniques to accurately capture the dynamics of the motor, inverter, and controller. The methodology ensures high fidelity while achieving computational efficiency for real-time applications. Validated using experimental and real-world data, the hybrid DT model demonstrated robust performance in reflecting the physical system's behavior and was successfully deployed on a real-time target machine. This deployment emphasized its suitability for real-time applications, showcasing its computational efficiency, accuracy, and ability to seamlessly integrate with real-world data streams.

By establishing a scalable and adaptable DT framework, this research contributes to advancing the field of DTs for EV propulsion systems, offering a valuable reference for future developments in autonomous and software-defined vehicle technologies.

Lühikokkuvõte

Autonoomse elektrisõiduki veoelektriajami digitaalse kaksiku täiustatud modelleerimise raamistikud

Tänapäeva kasvav nõudlus tõhusate, töökindlate ja jätkusuutlike elektrisõidukite (EV) järele on oluliselt edendanud uuenduslike tehnoloogiate arengut. Ühe uuendusliku tööriistana on selle käigus esile kerkinud digitaalkaksiku (DT) tehnoloogia. DT on füüsilise süsteemi reaalajas toimiv virtuaalne koopia, võimaldades selle töö jälgimist, analüüsimist ja optimeerimist kogu elutsükli vältel. Elektrisõidukite jõuülekandesüsteemide puhul pakub DT tehnoloogia võimalusi süsteemi jõudluse parandamiseks, ennetava hoolduse planeerimiseks ning süsteemi kohandamiseks, täiustades traditsioonilisi modelleerimis- ja simulatsioonimeetodeid.

Käesolev doktoritöö pakub põhjaliku raamistiku elektrisõiduki jõuülekandesüsteemi DT väljatöötamiseks, käsitledes väljakutseid, mis tulenevad süsteemi keerukusest, määramatusest ning osaliselt tundmatutest süsteemiparameetritest. Töö alguses antakse süsteemne ülevaade DT-de arengust ja nende rakendamisest elektrisõidukitest, tuues esile enim kasutatavad platvormid ning DT tehnoloogiate rakendused EV jõuülekandesüsteemides. Seejärel kirjeldatakse uuringu keskmes oleva “taaskasutatud” elektrisõiduki ISEAUTO jõuülekandesüsteemi, keskendudes eelkõige selle arhitektuurile ja unikaalsetele omadustele, mis on olulised modelleerimise aluse loomisel. Töö käigus analüüsitakse ka erinevaid modelleerimistehnikaid, selgitades iga meetodi valikukriteeriume ja sobivust erinevate komponentide modelleerimiseks. Mootori, inverteri ja kontrolleri dünaamika täpseks kirjeldamiseks pakutakse välja hübriidne lähenemine, mis ühendab nii füüsikalised kui ka andmepõhised modelleerimistehnikad. Välja pakutud metoodika tagab suure täpsuse ning arvutusliku efektiivsuse reaalajas töötavate rakenduste jaoks. Loodud hübriidne DT mudel valideeriti eksperimentaalsete ja reaalsete andmete põhjal ning rakendati füüsilise süsteemi peal. Mudel näitas häid tulemusi füüsilise süsteemi talitluse peegeldamisel, kinnitades selle sobivust reaalajas töötavate rakendustes ning tuues esile selle arvutusliku efektiivsuse, täpsuse ja võimekuse käsitleda füüsilise süsteemi andmevooge. Loodud skaleeritav ja kohaldatav DT raamistik annab olulise panuse elektrisõidukite jõuülekandesüsteemide DT-de valdkonna edendamisse. Käesolev doktoritöö pakub väärtusliku raamistiku autonoomsete ja tarkvarapõhiste sõidukite tehnoloogiate edasi arendamiseks.

Appendix 1

Publication I

Ibrahim, M.; Rassõlkin, A.; Vaimann, T.; Kallaste, A. Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System. Sustainability 2022, 14, 601. https://doi.org/10.3390/su14020601.

sustainability

MDPI

Review

Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System

Mahmoud Ibrahim *, Anton Rassõlkin, Toomas Vaimann and Ants Kallaste

Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, 19086 Tallinn, Estonia; anton.rassolkin@taltech.ee (A.R.); toomas.vaimann@taltech.ee (T.V.); ants.kallaste@taltech.ee (A.K.)
* Correspondence: mahmoh@taltech.ee

Abstract: The significant progress in the electric automotive industry brought a higher need for new technological innovations. Digital Twin (DT) is one of the hottest trends of the fourth industrial revolution. It allows representing physical assets under various operating conditions in a low-cost and zero-risk environment. DTs are used in many different fields from aerospace to healthcare. However, one of the perspective applications of such technology is the automotive industry. This paper presents an overview of the implementation of DT technology in electric vehicles (EV) propulsion drive systems. A general review of DT technology is supplemented with main applications analysis and comparison between different simulation technologies. Primary attention is given to the adaptation of DT technology for EV propulsion drive systems.

Keywords: electric vehicle propulsion drive system; digital twin; hardware in the loop; real-time simulation

check for updates

Citation: Ibrahim, M.; Rassõlkin, A.; Vaimann, T.; Kallaste, A. Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System. *Sustainability* **2022**, *14*, 601. https://doi.org/10.3390/su14020601

Academic Editors: J. C. Hernandez and Thanikanti Sudhakar Babu

Received: 25 November 2021
Accepted: 4 January 2022
Published: 6 January 2022

1. Introduction

Considerable values have been brought to the entire industry over the last decades due to digital manufacturing. Through virtually represented factories, resources, workforces, and skills, etc., digital manufacturing builds models and simulates product and process development. The remarkable progress in communication and information technologies has advanced the development of manufacturing widely [1]. Computer-aided technologies, including Computer-Aided Design (CAD), Computer-Aided Engineering (CAE), Computer-Aided Manufacturing (CAM), Finite Element Analysis (FEA), Product Data Management (PDM), etc., are quickly developing and playing a vitally critical role in the modern industry [2,3]. Advanced data analytics and the Internet of Things (IoT) connectivity have increased the volume of data usable from manufacturing, healthcare, and smart city environments [4]. IoT environment, coupled with data analytics, provides an essential resource for predictive maintenance, fault detection, the future health of manufacturing processes, and smart city developments [5]. Digital Twin (DT) can overcome integration between IoT and data analytics through its ability to create connected physical and virtual models. A DT environment enables high-speed and real-time simulation analysis accurately [6].

This review highlights DT as a trending technology in different applications and sectors as it is ongoingly discussed in the following sections. A deductive comparison between different simulation technologies over time is discussed in Section 1.1. Different existing and prospective applications of DT are presented in Section 1.2. In Section 1.3, varieties of DT software and platforms and their specific applications are discussed. DT for AEV propulsion drive system as the main review topic is extensively discussed in Section 2. A comparative analysis between Hardware in loop HIL and DT simulations for AEV propulsion drive systems is discussed in Sections 2.2 and 2.3, respectively. Figure 1 provides an illustrative diagram of the Introduction section's content.

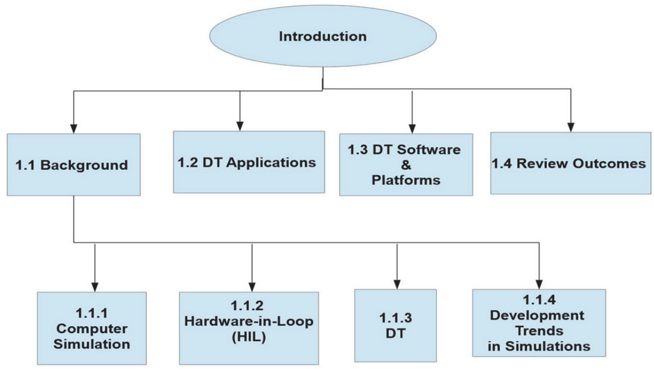


Figure 1. Introduction content diagram.

1.1. Background

Simulation history dates back to World War II when two mathematicians Jon Von Neumann and Stanislaw Ulam were puzzled by the behavior of neutrons. The problem was complicated; the hit and trial methods were too costly for them. They suggested the roulette wheel method at that time. The basic data regarding the occurrence of various events were known, into which the probabilities of separate events were merged in a step-by-step analysis to predict the outcome of entire sequence of events. Their technique had remarkable success on the neutron problem, and soon it became more popular and applicable in many business and industry applications [7]. In [8], Shannon described simulation as "the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behavior of the system or of evaluating various strategies (within limits imposed by a criterion or set of criteria) for the operation of the system".

1.1.1. Computer Simulation

The use of a computer is to represent the dynamic responses of a system by the behavior of another system modeled after it. A simulation uses a mathematical description, or model, of a real system in the form of a computer program. This model is composed of equations that duplicate the functional relationships within the real system. When the program is run, the resulting mathematical dynamics form an analog of the behavior of the real system, with the results presented in the form of data. For example, an electric machine can be described by a mathematical model that incorporates variables such as current, voltage, and magnetic flux. Additional mathematical equations can then be used to adjust the model to changes in certain variables, such as the winding material that is used to define heat dissipation losses.

A simulation can also take the form of a computer-graphics image that represents dynamic processes in an animated sequence, but some drawbacks pervade this technology, such as the following [9]:

- Mistakes may be made in the programming or rules of the simulation or model.
- Time may be needed to make sense of the results.
- No data exchange occurs between the real and the simulation models, which might limit the effectiveness of the results.
- People's reactions to the model or simulation might not be realistic or reliable.

1.1.2. Hardware-in-the-Loop (HIL)

At some point, thorough and reliable tests are necessary to verify and validate the design. However, as modern systems grow in complexity, particularly in software, this critical step is more easily said than performed [10]. The need for real-time simulation tools was necessary to overcome the problems concerning conventional computer simulations.

Hardware-in-the-Loop (HIL) simulation is a technique where physical signals from a controller are connected to a test system that physically simulates the situation, tricking the controller into thinking it is in the assembled product. Test and design iteration takes place as though the real-world system is being used, and one can easily run through thousands of possible scenarios to properly exercise your controller without the cost and time associated with actual physical tests. HIL helps to test the behavior of your control algorithms without physical prototypes. It is especially useful when testing your control algorithm on a real physical system is costly or dangerous. HIL simulation is widely used in the automotive, aerospace and defense, and industrial automation and machinery industries to test embedded designs. HIL is also being adopted in medical devices, communications, semiconductors, and other industries. For HIL to be of value, the quality of the simulation software is of utmost importance. Simulation software must be paired with hardware that not only accounts for system specifications such as connector type and I/O but also allows for fault insertion and the ability to test real-world scenarios [11]. A simple example of a HIL simulation system is an EV motor control unit MCU. The motor control unit MCU is responsible for converting sensor measurements into action such as adjusting the inverter frequency when the accelerator is depressed. A HIL test replaces the motor with a simulation comprising hardware and software that interacts with real I/O as though the physical motor was present. Due to the fact that updates can be made in software, you can quickly incorporate MCU, test a wide breadth of relevant scenarios, and expand test coverage as needed to fearlessly and comprehensively test without risk to a physical system.

In addition to the slightly high cost of real-time simulators, the main drawbacks of HIL systems are the complexity of development and verification. Figure 2 shows the basic components of a HIL system.

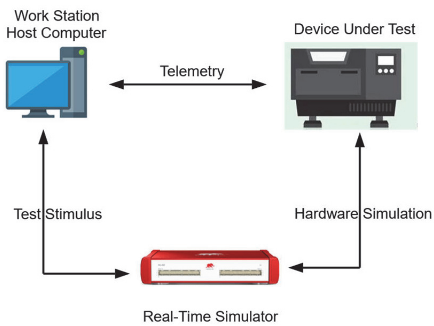


Figure 2. Hardware-in-the-Loop (HIL) System.

1.1.3. Digital Twins

Formal ideas around Digital Twins DT have been around since the early 2000s [2]. The concept of the first DT model was publicly introduced in 2002 by Grieves [12]. In early 2012, the first paper setting a key for DT definition was released by National Aeronautical Space Administration (NASA) [13]. They defined it as an integrated multiphysics, multiscale, and probabilistic simulation of a system that uses the best physical models, sensors, fleet history, etc., to simulate the life of its corresponding physical twin.

Chen [14] defined DT as a computerized model of a physical system that mirrors its functional features. Zheng et al. [15] said that DT is a virtual information set that describes actual physical assets. Mandi [16] defined DT as a virtual instance of a physical system (twin) continually updated with the latter's performance, maintenance, and health status data throughout the physical system's life cycle.

DTs also known as computational mega models mirror systems or avatars and can be defined as a digital representation of a physical object, process, or service. A digital twin can be a digital replica of an object in the physical world, such as a jet engine or wind farm, or even larger items such as buildings or even entire cities. In addition to physical assets, digital twin technology can be used to replicate processes in order to collect data to predict how they will perform.

A digital twin is, in essence, a computer program that uses real-world data to create simulations that can predict how a product or process will perform. These programs can integrate the IoT, artificial intelligence, and software analytics to enhance the output.

With the advancement of machine learning and factors such as Big Data, these virtual models have become a staple in modern engineering to drive innovation and improve performance [17].

A simple example explaining the DT of an object is a wind turbine outfitted with various sensors related to vital areas of functionality. These sensors produce data about different aspects of wind turbine performance, such as energy output, temperature, weather conditions, and more. These data are then relayed to a processing system and applied to the digital copy. Once informed with such data, the virtual model can be used to run simulations, study performance issues, and generate possible improvements, all to generate valuable insights—which can then be applied back to the original physical object. Figure 3 shows the main concept of the DT system.

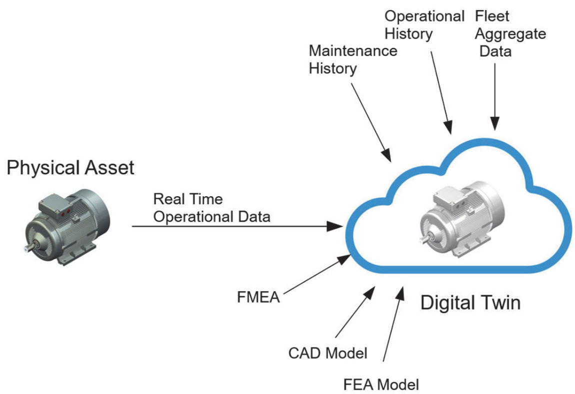


Figure 3. Digital Twin (DT) system.

At first glance, we can find common ground between DT and HIL simulations as they both fall under real-time simulations, but the essential difference between DTand HIL simulation is that for the latter one you build a software model of the core but have it interface with and direct real hardware (circuitry and mechanical) to assess the performance of your controller. For DT, you create a software-only model of the system being controlled and then provide it with inputs and outputs from the controller being tested and see how well your controller acts and whether it performs what it is supposed to be performing.

Although simulations and DTs both utilize digital models to replicate a system's various processes, a digital twin is actuallyl a virtual environment, which makes it considerably richer for study. The difference between digital twin and simulation is largely a matter of scale: While a simulation typically studies one particular process, a digital twin can itself run any number of useful simulations to study multiple processes. The differences do not end there. For example, simulations usually do not benefit from having real-time data. However, digital twins are designed around a two-way flow of information that first occurs when object sensors provide relevant data to the system processor and then happens again when insights created by the processor are shared back with the source object.

By having better and constantly updated data related to a wide range of areas, combined with the added computing power that accompanies a virtual environment, digital twins can study more issues from far more vantage points than standard simulations can—with greater ultimate potential to improve products and processes. In a word, Table 1 shows a brief comparison between discussed simulation technologies.

Table 1. Comparison between computer simulation, HIL, and DT.

	Computer Simulation	HIL	DT
Data element and interaction	Static	Active	Active
Simulation Basis	Potential parameters	Real time feedback	Real time feed back
Scope	Narrow-Primary design	Narrow-Advanced	Wide-Advanced

1.1.4. Types of Digital Twins

There are various types of digital twins depending on the level of product magnification. The biggest difference between these twins is the area of application [18].

Component twins/Parts twins: They are the basic unit of the DT and the smallest example of a functioning component. Parts twins are roughly the same thing but pertain to components of slightly less importance.

Asset twins: When two or more components work together, they form what is known as an asset. Asset twins allow you to study the interaction of those components, creating a wealth of performance data that can be processed and then turned into actionable insights.

System or Unit twins: The next level of magnification involves a system or unit twins, which enable you to observe how different assets come together to form an entire functioning system. System twins provide visibility regarding the interaction of assets and may suggest performance enhancements.

Process twins: The macro-level of magnification reveal how systems work together to create an entire production facility. They can help determine precise timing schemes that ultimately influence overall effectiveness.

1.1.5. Development Trends in Simulations

For decades, computer simulation tools were effective enough to answer specific design and modeling equations; however, by that time, they became limited due to the complexity of systems and the high amount of data being processed [19]. Proceeding to real-time, for a while, in-the-loop simulations could be a time, money, and effort saver by helping to identify errors before they occur in the target environment or at the customer. These simulations can be performed in various forms depending on the stage of the product development. Model-in-the-Loop (MIL), Software-in-the-Loop (SIL), Processor-in-the-Loop (PIL) simulation, and (HIL) are different forms of In-the-Loop simulations. There are proponents of DT who say HIL simulation is “so yesterday” and is no longer needed and proponents of HITL who claim that DTs are overhyped, oversold, and overly dependent on the fidelity of the model to reality [18]. Others opinions say that the best solution might be a hybrid of both DT and HIL. Figure 4 shows the timeline of the evolution of simulation technologies starting from the first individual simulation application and ending with DT technology.

1.2. Applications of Digital Twins

DTs’ first appearance was in industries of product and manufacturing design, then emerged in industries such as aerospace, automation, shipbuilding, healthcare, and smart cities [20].

Manufacturing: relies on high-cost equipment that generates a high volume of data, which facilitates creating DTs, because manufacturers are always looking for a method to track and monitor products to save time and money. This is why DTs seek to make the most significant impact within this setting. The current growth is in line with the industry 4.0 concept, coined the fourth industrial revolution; this harnesses the connectivity of devices to make the concept of DT a reality for manufacturing processes [21–23]. DT has the potential to provide real-time status on machines’ performance as well as production line feedback.

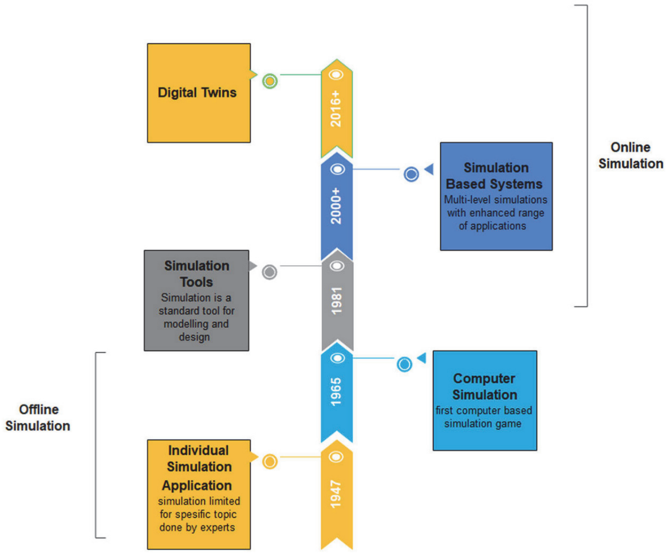


Figure 4. Evolution timeline of simulation technology.

Product development: is a long and intricate process. For instance, it takes up to 6 years to design and launch a new car model [24]. It needs to be a seamless transition from the preceding model to the new model. A slight mistake during the process can undermine the brand's value and profitability [21]. A DT helps to integrate data between previous-generation models with the new concept in their digital formats. Twinning also enables seamless communication between product designers, end customers, and other stakeholders. When it comes to product testing, having a DT negates the need to wait for performance data gathered during vehicle trials to determine its performance and quality [25].

Predictive maintenance: DT provides the manufacturer the ability to predict issues sooner, and their use increases connectivity and feedback between devices, in turn improving reliability and performance. AI algorithms coupled with DTs have the potential for greater accuracy as the machine can hold large amounts of data needed for performance and prediction analysis. DT is creating an environment to test products as well as a system that acts on real-time data; within a manufacturing setting, this has the potential to be a largely valuable asset [26,27].

Aerospace: Before DTs were found, physical twins were used in aerospace engineering. An example of this is the Apollo 13 program in the 1970s where NASA scientists on Earth were able to simulate the conditions of the ship and find answers when critical issues arose. Later in 2002, the DT concept is introduced by John Vickers from NASA [28]. Today, experts acknowledge the importance of DT in the aerospace sector where 75% of air force executives have cast the vote of confidence in favor of the digital twin, according to Business Wire's survey report [29]. With DTs, engineers can use predictive analytics to foresee any future problem involving airframes, engines, or other components to ensure the safety of the people on board.

Automotive: The development of an automotive is a long and complex process. Typically, new car model manufacturing might take five to six years—from the design stage to launch in the market [22]. The key to the success and long-term sustainability of an automotive organization is the effective design [30]. Even a small drawback in the product design can erode the company's brand value for a long time. With digital twin technology, it become easier to cover all phases of the automotive industry starting from design, development, monitoring, and maintenance of the vehicle. After the revolutionary development in batteries technology and the emergence of electric automobiles that place them on the list of global demand, digitization of automotive manufacturing and development process has become an urgent necessity [31].

Smart cities: have always relied on IoT technology for a while but with the increased number of smart cities, more connected communities are found; as a result, the need for new technologies such as DTs has increased. It can be used in planning new smart cities and help with ongoing developments of current smart cities [32]. There are also benefits within energy-saving as the collected data from IoT provides an excellent insight into how utilities are being distributed and used. In a digital twin city, the data of the operating status of infrastructure, the deployment of municipal resources, and the flow of people, logistics, and vehicles will be collected by sensors, cameras, and various digital subsystems. Modern communication technologies such as 5G are responsible for delivering data to the cloud and the city government to be monitored and processed; this makes the city more efficient [33].

Healthcare: The DT's provides researchers, doctors, hospitals, and healthcare providers the ability to simulate environments specific to their needs, whether it be real-time or looking to future developments and uses [34]. In addition to this, DTs can be used simultaneously with AI algorithms to make smarter predictions and decisions. Many applications within healthcare do not directly include the patient but are beneficial for ongoing care and treatment, hence the key role such systems have on patient care. DT for healthcare is in its infancy, but its potential is vast, from using it for bed management to large-scale wards and hospital management. Posessing the ability to simulate and act in real-time is even more paramount within healthcare as it can be the difference between life and death. DT could also assist with predictive maintenance and ongoing repair of medical equipment. The DT within the medical environment has the potential, along with Artificial Intelligence (AI), to make life-saving decisions based on real-time and historical data [35].

Ocean: Sustained ocean observations are an essential part of worldwide efforts to understand and protect marine ecological systems. Observation processes could be samples collected on ships; measurements from instruments on fixed platforms; autonomous and drifting systems; submersible platforms; and remote observing systems such as satellites and aircraft. Previously, ocean observation was a complex process and collecting data from it takes a long time and excessive cost because of different standards, nomenclature, and baselines. Digital Twin for oceans was a turning point that integrated a wide range of data sources, modeling and simulation, AI algorithms, and specialized tools including relevant best practices. DT forms a new globally shared capacity to access, manipulate, analyze, and visualize marine information. It enables users and partners to create ocean-related development scenarios, addressing issues such as green energy developments (renewable and non-renewable), mining impacts, fisheries and mariculture, marine protected area sitting, nature-based solutions, and ocean-based tourism. European Union (EU) was the first to take the initiative of investing in DT technology for the ocean by launching many projects in several member countries. Blue-Cloud is one of the ocean DT projects released by the EU aiming to the integrate all European assets related to seas and oceans with top-tier digital technologies into a digital component representing a consistent high-resolution, multi-dimensional, and (nearly) real-time description of the ocean [36].

Construction: As technology becomes more pervasive and smart buildings and precincts develop, real estate companies have tended to use their smarts to anticipate both customer and technological needs. A good method for performing this and to cut time and cost is digitalization. This technologically enabled process can deliver greater strategic value for the real estate industry as a whole. DTs can optimize operations and improve customer experience, and a twin can also deliver benefits across the full lifecycle of a building by simulating complex scenarios [37]. Buildings as a complex, high-value asset present an ideal opportunity for realizing the benefits of a DT. The full construction process can be planned, visualized, and optimized before the ground is even broken. Construction sites can be managed more effectively, with the ability to predict exactly how delays and decisions will impact overall construction. Moreover, the ability to monitor safety and compliance in real-time can save lives by predicting emergencies before they occur [38].

Figure 5 summarizes the applications discussed in this subchapter. In short, although DTs invaded many sectors, they are still new and not sufficiently covered for other applications, as is the case of EV propulsion drive systems, which will be addressed in the next section. The future of DT is nearly limitless due to the fact that increasing amounts of cognitive power are constantly being devoted to their use. Thus, DTs are constantly gaining new skills and capabilities, which means they can continue to generate insights needed to make products better and processes more efficient.

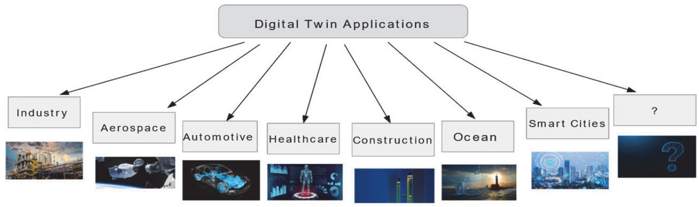


Figure 5. Digital Twin applications map.

1.3. Digital Twins Software and Platforms

DT requires advanced software able to generate a digital simulation of a physical entity. DT software is designed to monitor asset performance as well as to run simulations to predict potential outcomes or maintenance that might face the asset. There are many software and platforms from different companies that support DT simulation that are more suitable for engineering applications, such as the following:

Azure Digital Twin: Microsoft's platform enables the creation of twin graphs based on digital models of entire environments, which could be buildings, factories, farms, energy networks, railways, stadiums, and even entire cities. These digital models can be used to gain insights that drive better products, optimized operations, reduced costs, and breakthrough customer experiences.

AWS IoT: Amazon's platform. It's mainly used for remotely monitoring as it enables exchanging data and information between a remote emulation or simulation and the physical twin. AWS brings AI and IoT together to make devices more intelligent. You can create models in the cloud and deploy them to devices where they run two-times faster compared to other offerings.

Giraffe: A DT tool is used for construction applications. It accelerates the ability to scale by conducting site analysis in real-time with building design. Giraffe enables overlaying data, querying, and automatically calculating the proof of concept.

Perdix Platform: General Electric's platform. It is a complete solution for industrial monitoring and event management. This platform delivers shared capabilities that industrial applications require: asset connectivity, edge technologies, analytics, machine learning, Big Data processing, and asset-centric digital twins.

ETAP ADMS: Advanced Distribution Management System (ADMS) is a flexible solution for addressing the core requirement of the new digital grid to provide resiliency and reliability to the network. It provides an intelligent and robust decision support platform based on a unified Digital Twin of the electrical network with a collection of Geospatial-based distribution network applications integrated with mission-critical operational solutions to reliably and securely manage, control, visualize, and optimize small to vast distribution networks and smart grids.

Ansys Twin Builder: It is a platform that allows engineers to create simulation-based digital twins-digital representations of assets with real-world sensor inputs. It is mainly used for industrial applications for design, test, predictive maintenance, and optimization. Ansys Twin Bulder has different sub-platforms for each usage and different industrial applications.

Digital-Twin-Distiller: A python-based platform for DT simulation suitable for manufacturing applications. It allows researchers to develop and deploy simulation models. It aims to link research and engineering work environments to preserve simulation validation [39].

1.4. Review Outcomes

From the previous context, the trended technology of DT was addressed from different sides. The comparison between computer simulations, HIL, and DT highlighted the advantages of the last one. DTs receive real-time updates from the physical asset, process, or system. Therefore, the tests, assessments, and analysis work conducted by engineers are based on real-world conditions. As the state of the digital twin dynamically changes as it receives new data from the physical world, it matures, producing outputs that are more accurate and valuable. DTs can provide engineers with virtual tools that allow them to look at, explore, and assess physical assets, processes, and systems. With this ability, it is possible to obtain an accurate view of what is happening now, as well as what will happen in the future. Many applications have been well covered by DT technology but some still in the early stages, which will be discussed in the next section. Many companies have developed software and platforms to keep pace with DT technology in line with their products.

2. Overview of Recent Trends in EV Drive Systems

2.1. Background

As it can be observed from the literature review in the previous section, the automotive industry is one of the top existing applications of DT; however, research studies concerning Electric Vehicles EV propulsion drive systems are so limited and in need of being studied deeply.

Electric vehicles (EV) include fully (battery) and hybrid EVs in continuous growth by the time, and in 2020 it increased by 43% more than in 2019 [40]. Information Handling Services (IHS) Markit predicts that EVs could capture 45% of the new car market already in 2040 and nearly 80% by 2050 due to great technological advances, decreased manufacturing costs, and international policies that facilitate EV expansion. Figure 6 shows a comparison between EVs, including battery (BEV) and plug-in hybrid (PHEV), fuel cell (FCV), and traditional internal combustion engine (ICE) vehicles [41].

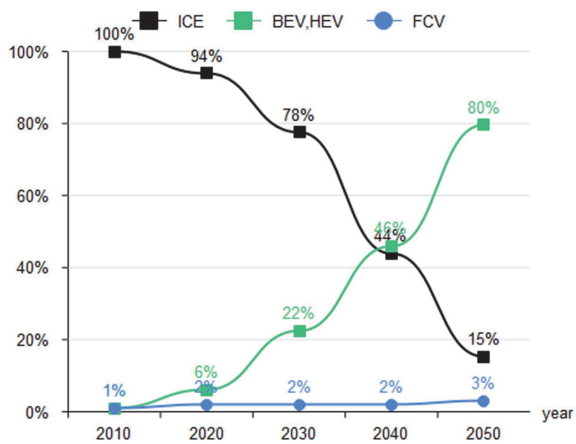


Figure 6. Comparison between global EV, ICE, and FCV markets.

EV propulsion drive system is considered the core element of the vehicle. It needs to be efficient, reliable, and economically sufficient to yield satisfactory EV operation performance [42]. EV propulsion drive system comprises both mechanical and electrical parts, as shown in Figure 7. In this review, the electrical parts are the main topics of focus.

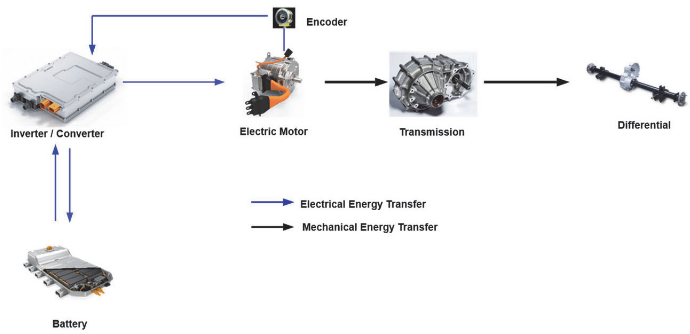


Figure 7. Electrical Vehicle (EV) drive system components.

Battery: Many research studies considered the battery as a component of the EV drive system, including its attached heating and management systems [43]. The maximum driving distance of an EV is often determined by the battery's capacity—the higher the capacity, the higher the driving distance. Lithium-ion batteries (LIBs), Lithium Nickel Manganese-Cobalt (NMC) oxide, and Lithium Nickel-Cobalt-Aluminium (NCA) oxide are dominating the EV battery industry with nearly 96% of market share in 2019 [44,45]. The EV battery has a direct impact on inverter design and operation. DT technology can be investigated in EV battery health monitoring, faults detection, and lifetime prediction.

Inverter/Converter: The power electronics component of the drive system. It comprises three sub-components: DC-DC converter, inverter, and motor control unit MCU. The main DC-DC Converter converts the battery high voltage DC into low voltage DC to power headlights, interior lights, wiper and window motors, fans, pumps, and many other systems.

The inverter includes a motor control unit (MCU) that is usually an integrated unit. An inverter converts the battery's high DC voltage into AC variable frequency voltage, which is then used to regulate the motor speed. MCU implements the control algorithm of the EV motor. It configures motor speed and torque after receiving comments from the vehicle control unit (VCU) via CAN-bus communication and then translates them into power signals by the inverter to be fed to the motor. The inverter is responsible for executing acceleration and deceleration, which is crucial in maximizing the EV's drivability. During vehicle braking, it can regenerate DC power back to the battery for charging. It is a very sensitive part of the EV as it is the focal point between the stationary energy and the kinetic energy part of the EV [46,47]. Silicon (Si) insulated gate bipolar transistor (IGBT) was widely used in EV drive inverters since 1980. It has the combined advantages of the simple gate-drive of a field-effect transistor and the high current, low conduction loss of bipolar transistor. IGBTs can block high voltages with low on-state conduction losses and well-controlled switching times. However, they are limited by how fast they can switch while delivering low on-state conduction losses. This results in a need for costly and large-size thermal-management methods and a limitation on power-conversion system efficiency. With the revolutionary progress in power electronics development such as SiC (silicon carbide) and GaN (gallium nitride), metal-oxide-semiconductor field-effect transistor (MOSFET)-based inverters are alternatives, while still possessing higher prices; they provide a better thermal profile, lower switching losses, higher efficiency, and longer lifetime [48]. Building a DT model of the EV inverter would be an ideal solution for EV manufacturers as well as researchers for development, health monitoring, fault detection, and also components' lifetime prediction.

Electric motor: is the base stone of the EV. It converts electric energy from the battery into kinetic energy that moves the wheels. The advantage of using the motor instead of an engine is numerous: first, the noise and the vibration. Many passengers riding EVs for the first time are surprised by how quiet and comfortable the ride feels. Moreover, the EV powertrain is smaller than the engine, thus providing additional space for efficient vehicle design—such as expanded cabin space or storage [49]. The motor is also in part an electric generator—it converts the kinetic energy generated while in neutral gear (e.g., while the car is proceeding downhill) into electric energy saved to the battery. There are five types of motors most often used in EVs: Permanent Magnet Synchronous Motors (PMSM) are the most used type by many manufacturers such as Hyundai and GM [50]. Brushless DC Motors (BLDC) are commonly used in light electric carts and electric scooters. Three-phase Induction Motors (IM) are widely used by many EV manufacturers such as Tesla [51]. Permanent Magnet Assisted Synchronous reluctance Motors (PMSynRM) recently started to be used in some Tesla models to alternate induction motors. Switched Reluctance Motors (SRM) started to gain higher interest recently, and some manufacturers such as BMW and LandRover are developing these types of motors to use in their EVs. [52].

2.2. Problem Definition

Developing optimal designs for EV drive systems is very challenging for many researchers and also EV manufacturers. Most of the studies start from computer simulation, specially designed environments. The next step is transitioning from simulation to real hardware building and testing. Usually, test benches are built to replicate simulated models, and comparative analysis between the results of the two models is approached. After building and running the test bench, optimization of the drive system components, faulty operations, and different malfunctions are normally tested. In performing this, firstly, depending on the power rating of the system, a lot of electrical energy is consumed with a direct impact on the price of the development of the system. On the other hand, testing of one system component sometimes can result in faulting another component. Moreover, testing faulty conditions requires intentionally faulting some components of the test bench. The following subsections will discuss solutions to overcome the above-mentioned issues.

2.3. Hardware in the Loop Simulation for EV Drive Systems

EV drive system design and installation are expensive and time consuming. Mainly, it consumes much time and financial resources to perform tests and debugging of the equipment. Usually, those tests are executed by constructing test rigs with real machines, but experimental designing is expensive and, in many cases, difficult to implement [53]. Thus, real-time simulators provide an efficient solution for those problems.

Mudrov et al. [54] presented a deductive study on using Power HIL (PHIL) systems for EV drive systems. They concluded that the main advantage of the PHIL system is the ability to simulate the electromechanical part behavior to take part of electrical drive power part and to make a think converter that it works with a real electrical drive. They also provided a PHIL system for EV drives based on a power converter with Field Programmable Gate Array (FPGA). The proposed system had the capability of testing a multistage inverter control system.

Poon et al. in [55] proposed a HIL platform for EV drive applications. The proposed platform can test the drive cycle and fault injection.

Berry and Gu [56] proposed a real-time HIL model of a three-phase EV power inverter system to simulate thermal behavior and internal losses using an FPGA real-time system. The presented model can accommodate any switching method of the inverter.

Collin et al. [57] provided an HIL prototype model of SiC-based drive system of PMSM. Typhoon 402 HIL module used as a system-level controller. The provided prototype was tested under Sinusoidal Pulse Width Modulation (SPWM) and Space Vector Pulse Width Modulation (SVPWM) techniques. The testbed showed the advantage of a faster switching time of the SiC device, which resulted in a total harmonic reduction in the motor current.

Mishra et al. in [58] proposed an HIL simulation model of a PMSM drive system. They used the Xilinx system generator platform coupled with the MATLAB simulation model of the drive system. They used different combinations of simulation environments to highlight the difference in system performance.

Amitkumar et al. [59] proposed a HIL simulation system to study the impact of PMSM inverter faults used for an EV drive system aiming to reduce the chance of equipment damage during testing. Three types of driving inverter gate-drive failure faults (device open-circuit fault) of one or more switches were studied. PMSM emulation system was implemented with its vector controller on an FPGA of a real-time simulator HIL to minimize controller sampling time.

From the above, it is clear to us that HIL simulation is an ideal solution to test the components of the electric propulsion system and simulate some of the malfunctions expected in it as well as provide an appropriate environment to apply optimization techniques to improve the performance of the system and achieve higher efficiency of its components, but there are aspects of shortcomings that are flawed, which is the inability to simulate the entire EV drive as one system, as well as the limited handling of artificial intelligence tools.

2.4. Digital Twin (DT) for EV Drive System

DTs for EV drive systems are frequently used for system health monitoring, diagnostics, prognostics, optimization, scenario, and risk assessment [2]. They can be created at the system level, subsystem level, individual component level, and many other assets. In this section, different approaches to DT technology used in EV drive systems will be discussed.

Wunderlich and Santi [60] proposed an approach for a real-time DT model of a power electronic converter. Based on a dynamic Nonlinear Autoregressive eXogenous neural network (NARX-ANN), combinations of time-domain, switch-averaged, large-signal, real-time, and embeddable models are used. A boost converter model with the current source was their physical model. The proposed DT model of the converter can run on any platform, including locally on the converter's digital controller. Their model is mainly used for fault detection, prognostics and health management, and scenario and risk assessment.

Rjabtšikov et al. in [61] proposed a fault detection DT model for an AC 3-phase IM. Inter-turn short circuit fault detection was implemented into the motor DT. The emulator was built based on historical data and a mathematical model of the motor using Unity 3D combined with ROS service to enable online condition monitoring. The DT model in this case study was used as a virtual sensor for the physical motor model.

Toso et al. in [62] applied the DT to EV motor aiming to optimize the motor performance concerning estimating driving torque and cooling control. Thermal and electromagnetic FEA of EV induction motor was provided at first to collect the necessary data for both optimization operations. DT model of the motor was built using a micro lab box as a system-level DT.

Katukula et al. in [63] provided a DT system to monitor and analyze IM conditions. The provided DT system measures the drawn current by the motor using sensors. The collected data are fed to a simulation FEM model. The proposed DT enables a better understanding of the motor's thermomagnetic behavior and allows the ability to predict possible faults.

Rassõlkin et al. in [64] provided a methodology of collecting required data from an autonomous EV loading motor drive system based on the empirical performance model than using the collected data to develop a DT model. The data collected concerned the estimation of the drive system's performance. Unity 3D was used as a host environment for simulation and visualization of the motor DT model.

Brandtstaedter et al. in [65] presented DT model for fault detection. A 50 MW PMSM of an electric drive train was numerically simulated, and the framework of fault identification was presented. Unbalanced detection and temperature prediction in the rotor system were tested and verified using the digital model.

Jitong et al. in [66] presented a DT model of a three-phase IM. The provided model is built at first to realize the motor design then to monitor the motor equipment's normal operation to detect the entire lifecycle and shorten maintenance time. A 3D simulation model of the physical motor was built using 3D Max. Based on Unity 3D software, the digital twin was built depending on the provided simulation model. Data acquisition between real and digital models was made using an SQL server.

Venkatesan et al. [67] provided a DT system of an EV-PMSM drive system for health monitoring and prognosis purposes such as outputs casing temperature, winding temperature, time to refill the bearing lubricant, and percentage deterioration of magnetic flux to compute remaining useful life (RUL) of permanent magnet (PM). They presented two approaches for motor health monitoring: one to monitor the motor performance in-house and the other remotely. The provided DT model was built in MATLAB/Simulink, with Artificial Neural Network (ANN) and fuzzy logic to map the system inputs.

From the above analysis, it could be observed that DTs offer advanced solutions in the development of EV drive systems due to their ability to exchange a huge amount of data with the real model in no time and their ability to represent components, assets, or the entire system. DTs also have the possibility of working for different purposes such as prognosis, fault detection, health monitoring, lifetime prediction, and optimization. Table 2 provides a comparison between HIL simulation technology and DT for AEV drive systems.

Table 2. Comparison between HIL and DT technologies for EV drive systems.

Points of Comparison	Hardware-in-Loop (HIL)	Digital Twin (DT)
Simulation	Real-time (Online)	Real-time (Online)
Applications	Design; Testing; Optimization; Fault Detection	Design; Diagnosis; Optimization; Predictive Maintenance; Fault Detection; Health Monitoring; Life Time Prediction
Areas of Applications	Component; Subsystem	Component Subsystem; Whole System
Cost	High	Moderate
Reliability	High	Very High

3. Conclusions and Future Work

3.1. Conclusions

An extensive overview of DT technology has been provided. Compared to previous simulation technology, it is a powerful alternative and a major development in the topic of digital simulation and the connection between the virtual and physical worlds. DT is already applied in many different applications such as industry, aerospace, automotive, healthcare, and oceans. They offer a lot of functions such as health monitoring, fault detection, optimization, prognosis, and lifetime prediction. Although automotive is one of the current DT applications, most researchers care about vehicle design, motion, and visualization. Concerning EV propulsion drive system, it is still too new of a topic to say it is covered by DT technology. The analytical comparison between using HIL simulations and DTs for EV propulsion drive systems resulted in the following: HIL simulation is suitable for testing a drive system component or at most an asset. It is recommended to be applied in the phase of designing or testing the performance of the system. It also might be used for some common fault detection. DT is more effective after obtaining the physical drive system model. Due to their great ability to deal with a huge amounts of different datasets, DTs can be built for a component, asset, or the entire drive system. They can also be adapted for multiple uses, such as predictive maintenance, fault detection, health monitoring, and lifetime prediction, depending on the model basis and the type of data exchanged with the physical model. DTs can also be built to optimize the performance of the drive system that might be suitable for research and development purposes.

3.2. Future Works

A combination of HIL simulations and DTs is the best solution depending on the stage of the work, such as the following:

1. For EV propulsion drive system design, the first stage is to build simulation models of different system components;
2. Verifying different models using HIL simulation tests;
3. Building EV propulsion drive system physical models (test bench);

4. Building DT models of the EV drive system starting with components, then assets, and finally the entire system model. DT models will be built based on previously collected data from HIL simulation tests and the current exchanged data with the physical model.
5. Various datasets from the physical EV drive system can be obtained by the implementation of different sensors and data acquisition platforms. The DT model may include regulation for both virtual and physical entities that can be used for maintenance, diagnostic, optimization, and system development.

Author Contributions: Conceptualization, A.R. and M.I.; methodology, M.I.; formal analysis, T.V. and A.K.; investigation, M.I.; resources, M.I.; data curation, M.I.; writing—original draft preparation, M.I. and A.R.; writing—review and editing, T.V. and A.K.; visualization, M.I.; supervision, A.R.; funding acquisition, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of autonomous electric vehicle".

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [CrossRef]
2. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**, *8*, 108952–108971. [CrossRef]
3. Mandolla, C.; Petruzzelli, A.M.; Percoco, G.; Urbinati, A. Building a digital twin for additive manufacturing through the exploitation of blockchain: A case analysis of the aircraft industry. *Comput. Ind.* **2019**, *109*, 134–152. [CrossRef]
4. Raja, H.A.; Vaimann, T.; Rassolkin, A.; Kallaste, A.; Belahcen, A. IoT Based Tools for Data Acquisition in Electrical Machines and Robotics. In Proceedings of the 2021 IEEE 19th International Power Electronics and Motion Control Conference, PEMC, Gliwice, Poland, 25–29 April 2021; pp. 737–742. [CrossRef]
5. Bilberg, A.; Malik, A.A. Digital twin driven human–robot collaborative assembly. *CIRP Ann.* **2019**, *68*, 499–502. [CrossRef]
6. Schleich, B.; Anwer, N.; Mathieu, L.; Wartzack, S. Shaping the digital twin for design and production engineering. *CIRP Ann.* **2017**, *66*, 141–144. [CrossRef]
7. Goldsman, D.; Nance, R.E.; Wilson, J.R. A brief history of simulation. In Proceedings of the Winter Simulation Conference, Austin, TX, USA, 13–16 December 2009; pp. 310–313. [CrossRef]
8. Wimo, A.; Handels, R.; Jönsson, L. The art of simulation. *Lancet Healthy Longev.* **2020**, *1*, e2–e3. [CrossRef]
9. Lahiri, S.K. Offline Simulation. *Multivariable Predict. Control* **2017**, 167–182. [CrossRef]
10. Nguyen, V.H.; Tran, Q.T.; Guillo-Sansano, E.; Kotsampopoulos, P.; Gavriluta, C.; Lauss, G.; Strasser, T.I.; Jensen, T.V.; Heussen, K.; Gehrke, O.; et al. Hardware-in-the-Loop Assessment Methods. In *European Guide to Power System Testing*; Springer: Cham, Switzerland, 2020; pp. 51–66. [CrossRef]
11. Nabi, S.; Balike, M.; Allen, J.; Rzemien, K. An overview of hardware-in-The-loop testing systems at Visteon. *SAE Tech. Pap.* **2004**. [CrossRef]
12. San, O. The digital twin revolution. *Nat. Comput. Sci.* **2021**, *1*, 307–308. [CrossRef]
13. Glaessgen, E.H.; Stargel, D.S. The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. In Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Honolulu, HI, USA, 23–26 April 2012.
14. Chen, Z.; Huang, L. Digital twins for information-sharing in remanufacturing supply chain: A review. *Energy* **2020**, *220*, 119712. [CrossRef]
15. Zheng, Y.; Yang, S.; Cheng, H. An application framework of digital twin and its case study. *J. Ambient Intell. Humaniz. Comput.* **2018**, *10*, 1141–1153. [CrossRef]
16. Madni, A.M.; Madni, C.C.; Lucero, S.D. Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems* **2019**, *7*, 7. [CrossRef]
17. Wu, J.; Yang, Y.; Cheng, X.; Zuo, H.; Cheng, Z. The Development of Digital Twin Technology Review. In Proceedings of the 2020 Chinese Automation Congress, CAC, Shanghai, China, 6–8 November 2020; pp. 4901–4906. [CrossRef]
18. Singh, M.; Fuenmayor, E.; Hinchy, E.P.; Qiao, Y.; Murray, N.; Devine, D. Digital Twin: Origin to Future. *Appl. Syst. Innov.* **2021**, *4*, 36. [CrossRef]
19. Weyer, S.; Meyer, T.; Ohmer, M.; Gorecky, D.; Zühlke, D. Future Modeling and Simulation of CPS-based Factories: An Example from the Automotive Industry. *IFAC-PapersOnLine* **2016**, *49*, 97–102. [CrossRef]
20. Liu, M.; Fang, S.; Dong, H.; Xu, C. Review of digital twin about concepts, technologies, and industrial applications. *J. Manuf. Syst.* **2020**, *58*, 346–361. [CrossRef]

21. Wagner, S.; Milde, M.; Barhebwa-Mushamuka, F.; Reinhart, G. *Digital Twin Design in Production*; Springer: Cham, Switzerland, 2021. [CrossRef]
22. Lu, Y.; Liu, C.; Wang, K.I.-K.; Huang, H.; Xu, X. Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robot. Comput. Manuf.* **2020**, *61*, 101837. [CrossRef]
23. Stark, R.; Fresemann, C.; Lindow, K. Development and operation of Digital Twins for technical systems and services. *CIRP Ann.* **2019**, *68*, 129–132. [CrossRef]
24. Elsbach, K.D.; Stigliani, I. Design Thinking and Organizational Culture: A Review and Framework for Future Research. *J. Manag.* **2018**, *44*, 2274–2306. [CrossRef]
25. Jiang, Y.; Yin, S.; Li, K.; Luo, H.; Kaynak, O. Industrial applications of digital twins. *Philos. Trans. R. Soc. A* **2021**, *379*, 20200360. [CrossRef]
26. Centomo, S.; Dall'Ora, N.; Fummi, F. The Design of a Digital-Twin for Predictive Maintenance. In Proceedings of the IEEE Symposium on Emerging Technologies and Factory Automation, ETFA, Vienna, Austria, 8–11 September 2020; pp. 1781–1788. [CrossRef]
27. Zhang, C.; Xu, W.; Liu, J.; Liu, Z.; Zhou, Z.; Pham, D.T. A Reconfigurable Modeling Approach for Digital Twin-based Manufacturing System. *Procedia CIRP* **2019**, *83*, 118–125. [CrossRef]
28. Phanden, R.K.; Sharma, P.; Dubey, A. A review on simulation in digital twin for aerospace, manufacturing and robotics. *Mater. Today Proc.* **2021**, *38*, 174–178. [CrossRef]
29. Yin, H.; Wang, L. Application and Development Prospect of Digital Twin Technology in Aerospace. *IFAC-PapersOnLine* **2020**, *53*, 732–737. [CrossRef]
30. Gyimesi, K.; Berman, S. IBM Global Business Services Digital Transformation in the Automotive Industry Creating New Business Models where Digital Meets Physical. May 2011. Available online: https://www.ibm.com/downloads/cas/LVDZDXOA. (accessed on 2 January 2022).
31. Rahman, I.; Tadayoni, R. Digital Transformation of Automobiles-From product to service. In Proceedings of the 11th CMI International Conference: Prospects and Challenges Towards Developing a Digital Economy within the EU, PCTDDE, Copenhagen, Denmark, 29–30 November 2018; pp. 7–13. [CrossRef]
32. White, G.; Zink, A.; Codecá, L.; Clarke, S. A digital twin smart city for citizen feedback. *Cities* **2021**, *110*, 103064. [CrossRef]
33. Mohammadi, N.; Taylor, J.E. Smart city digital twins. In Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence, SSCI, Honolulu, HI, USA, 27 November–1 December 2017; pp. 1–5. [CrossRef]
34. Karakra, A.; Fontanili, F.; Lamine, E.; Lamothe, J. HospiT'Win: A predictive simulation-based digital twin for patients pathways in hospital. In Proceedings of the 2019 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI, Chicago, IL, USA, 19–22 May 2019. [CrossRef]
35. Erol, T.; Mendi, A.F.; Dogan, D. The Digital Twin Revolution in Healthcare. In Proceedings of the 4th International Symposium on Multidisciplinary Studies and Innovative Technologies, ISMSIT, Istanbul, Turkey, 22–24 October 2020. [CrossRef]
36. Barbie, A.; Pech, N.; Hasselbring, W.; Flogel, S.; Wenzhofer, F.; Walter, M.; Shchekinova, E.; Busse, M.; Turk, M.; Hofbauer, M.; et al. Developing an Underwater Network of Ocean Observation Systems with Digital Twin Prototypes-A Field Report from the Baltic Sea. *IEEE Internet Comput.* **2021**. [CrossRef]
37. Opoku, D.-G.J.; Perera, S.; Osei-Kyei, R.; Rashidi, M. Digital twin application in the construction industry: A literature review. *J. Build. Eng.* **2021**, *40*, 102726. [CrossRef]
38. Boje, C.; Guerriero, A.; Kubicki, S.; Rezgui, Y. Towards a semantic Construction Digital Twin: Directions for future research. *Autom. Constr.* **2020**, *114*, 103179. [CrossRef]
39. Orosz, T.; Gadó, K.; Katona, M.; Rassõlkin, A. Automatic Tolerance Analysis of Permanent Magnet Machines with Encapsuled FEM Models Using Digital-Twin-Distiller. *Processes* **2021**, *9*, 2077. [CrossRef]
40. Sanguesa, J.A.; Torres-Sanz, V.; Garrido, P.; Martinez, F.J.; Marquez-Barja, J.M. A Review on Electric Vehicles: Technologies and Challenges. *Smart Cities* **2021**, *4*, 372–404. [CrossRef]
41. Anosike, A.; Loomes, H.; Udokporo, C.K.; Garza-Reyes, J.A. Exploring the challenges of electric vehicle adoption in final mile parcel delivery. *Int. J. Logist. Res. Appl.* **2021**, 1–25. [CrossRef]
42. Kumar, L.; Jain, S. Electric propulsion system for electric vehicular technology: A review. *Renew. Sustain. Energy Rev.* **2014**, *29*, 924–940. [CrossRef]
43. Vaideeswaran, V.; Bhuvanesh, S.; Devasena, M. Battery Management Systems for Electric Vehicles using Lithium-Ion Batteries. In Proceedings of the 2019 Innovations in Power and Advanced Computing Technologies, i-PACT, Vellore, India, 22–23 March 2019. [CrossRef]
44. Salgado, R.M.; Danzi, F.; Oliveira, J.E.; El-Azab, A.; Camanho, P.P.; Braga, M.H. The Latest Trends in Electric Vehicles Batteries. *Molecules* **2021**, *26*, 3188. [CrossRef]
45. Katari, J.S.; Sneha, R.; Vinayaka, K.U.; Rana, K. A Concise review of different standards for performance testing of Lithium-ion Batteries for Electric Vehicle applications. In Proceedings of the 2020 IEEE International Conference on Power Systems Technology, POWERCON, Bangalore, India, 14–16 September 2020. [CrossRef]
46. Reimers, J.; Dorn-Gomba, L.; Mak, C.; Emadi, A. Automotive Traction Inverters: Current Status and Future Trends. *IEEE Trans. Veh. Technol.* **2019**, *68*, 3337–3350. [CrossRef]

47. Chi-Lan, C.; Xiao-Gang, W.; Yue-Wei, B.; Yan-Chun, X.; Kai, L. Key Technologies of EV Motor Drive System Design. *Procedia Eng.* **2011**, *16*, 356–362. [CrossRef]
48. Ji, S.; Zhang, Z.; Wang, F. Overview of high voltage sic power semiconductor devices: Development and application. *CES Trans. Electr. Mach. Syst.* **2020**, *1*, 254–264. [CrossRef]
49. Bhatt, P.; Mehar, H.; Sahajwani, M. Electrical Motors for Electric Vehicle–A Comparative Study. *SSRN Electron. J.* **2019**. [CrossRef]
50. Momen, F.; Rahman, K.M.; Son, Y.; Savagian, P. Electric Motor Design of General Motors' Chevrolet Bolt Electric Vehicle. *SAE Int. J. Altern. Powertrains* **2016**, *5*, 286–293. [CrossRef]
51. Sieklucki, G. An investigation into the induction motor of tesla model S vehicle. In Proceedings of the 2018 International Symposium on Electrical Machines, SME, Andrychow, Poland, 10–13 June 2018. [CrossRef]
52. Goli, C.S.; Manjrekar, M.; Essakiappan, S.; Sahu, P.; Shah, N. Landscaping and review of traction motors for electric vehicle applications. In Proceedings of the 2021 IEEE Transportation Electrification Conference and Expo, ITEC, Chicago, IL, USA, 21–25 June 2021; pp. 162–168. [CrossRef]
53. Mudrov, M.; Ziuzev, A.; Nesterov, K.; Valtchev, S. Power electrical drive Power-Hardware-in-the-Loop system: 2018 X International Conference on Electrical Power Drive Systems (ICEPDS). In Proceedings of the 2018 X International Conference on Electrical Power Drive Systems (ICEPDS), Novocherkassk, Russia, 3–6 October 2018. [CrossRef]
54. Mudrov, M.; Klimov, G.; Tecle, S.I.; Ziuzev, A. Experience in Using Real Time Simulators in Variable-Frequency Drives. In Proceedings of the 2020 XI International Conference on Electrical Power Drive Systems (ICEPDS), St. Petersburg, Russia, 4–7 October 2020. [CrossRef]
55. Poon, J.J.; Kinsy, M.A.; Pallo, N.A.; Devadas, S.; Čelanović, I.L. Hardware-in-the-loop testing for electric vehicle drive applications. In Proceedings of the 2012 Twenty-Seventh Annual IEEE Applied Power Electronics Conference and Exposition-APEC, Orlando, FL, USA, 5–9 February 2012; pp. 2576–2582. [CrossRef]
56. Berry, D.J.; Gu, E. FPGA based hardware in the loop inverter loss and thermal model for electric vehicles. In Proceedings of the 2020 IEEE Transportation Electrification Conference and Expo, ITEC, Chicago, IL, USA, 23–26 June 2020; pp. 75–80. [CrossRef]
57. Collin, R.; Stephens, M.; von Jouanne, A. Development of SiC-Based Motor Drive Using Typhoon HIL 402 as System-Level Controller. In Proceedings of the 2020 IEEE Energy Conversion Congress and Exposition, Detroit, MI, USA, 11–15 October 2020; pp. 2689–2695. [CrossRef]
58. Mishra, I.; Nath Tripathi, R.N.; Kumar Singh, V.K.; Hanamoto, T. A Hardware-in-the-Loop Simulation Approach for Analysis of Permanent Magnet Synchronous Motor Drive. In Proceedings of the 2019 10th International Conference on Power Electronics and ECCE Asia (ICPE 2019-ECCE Asia), Busan, Korea, 27–30 May 2019.
59. Amitkumar, K.S.; Pillay, P.; Belanger, J. Hardware-in-the-loop Simulations of Inverter Faults in an Electric Drive System. In Proceedings of the 2019 IEEE Energy Conversion Congress and Exposition, ECCE, Baltimore, MD, USA, 29 September–3 October 2019; pp. 353–358. [CrossRef]
60. Wunderlich, A.; Santi, E. Digital twin models of power electronic converters using dynamic neural networks. In Proceedings of the 2021 IEEE Applied Power Electronics Conference and Exposition-APEC, Phoenix, AZ, USA, 14–17 June 2021; pp. 2369–2376. [CrossRef]
61. Rjabtsikov, V.; Rassolkin, A.; Asad, B.; Vaimann, T.; Kallaste, A.; Kuts, V.; Jegorov, S.; Stepien, M.; Krawczyk, M. Digital Twin Service Unit for AC Motor Stator Inter-Turn Short Circuit Fault Detection. In Proceedings of the 2021 28th International Workshop on Electric Drives: Improving Reliability of Electric Drives (IWED), Moscow, Russia, 27–29 January 2021. [CrossRef]
62. Toso, F.; Torchio, R.; Favato, A.; Carlet, P.G.; Bolognani, S.; Alotto, P. Digital twins as electric motor soft-sensors in the automotive industry. In Proceedings of the 2021 IEEE International Workshop on Metrology for Automotive, MetroAutomotive, Bologna, Italy, 1–2 July 2021; pp. 13–18. [CrossRef]
63. Tshoombe, B.K.; Dos Santos, J.F.; Araujo, R.C.F.; Fonseca, W.D.S. Implementation of DT-based monitoring system of induction motors. In Proceedings of the 2021 14th IEEE International Conference on Industry Applications (INDUSCON), São Paulo, Brazil, 15–18 August 2021; pp. 161–166. [CrossRef]
64. Rassolkin, A.; Rjabtsikov, V.; Vaimann, T.; Kallaste, A.; Kuts, V.; Partyshev, A. Digital Twin of an Electrical Motor Based on Empirical Performance Model. In Proceedings of the 2020 XI International Conference on Electrical Power Drive Systems (ICEPDS), St. Petersburg, Russia, 4–7 October 2020. [CrossRef]
65. Brandtstaedter, H.; Hübner, L.; Jungiewicz, A.; Ludwig, C.; Tsouchnika, E.; Wever, U. DIGITAL TWINS FOR LARGE ELECTRIC DRIVE TRAINS. In Proceedings of the 2018 Petroleum and Chemical Industry Conference Europe (PCIC Europe), Antwerp, Belgium, 5–7 June 2018.
66. Jitong, X.; Lin, W.; Yan, Z. Design and Development of Digital Twin System for Intelligent Motor. In Proceedings of the 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA, Dalian, China, 25–27 August 2020; pp. 991–996. [CrossRef]
67. Venkatesan, S.; Manickavasagam, K.; Tengenkai, N.; Vijayalakshmi, N. Health monitoring and prognosis of electric vehicle motor using intelligent-digital twin. *IET Electr. Power Appl.* **2019**, *13*, 1328–1335. [CrossRef]

Appendix 2

Publication II

Ibrahim, M.; Rjabtšikov, V.; Gilbert Zequera, R. A. Overview of Digital Twin Platforms for EV Applications. Sensors 2023, 23, 1414. https://doi.org/10.3390/s23031414.

sensors

MDPI

Review

Overview of Digital Twin Platforms for EV Applications

Mahmoud Ibrahim *, Viktor Rjabtšikov and Rolando Gilbert

Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, 19086 Tallinn, Estonia
* Correspondence: mahmoud.mohamed@taltech.ee

Abstract: Digital twin (DT) technology has been used in a wide range of applications, including electric vehicles. The DT platform provides a virtual representation or advanced simulation of a physical object in real-time. The implementation of DT on various aspects of EVs has recently transpired in different research studies. Generally, DT can emulate the actual vehicle on the road to predict/optimize its performance and improve vehicle safety. Additionally, DT can be used for the optimization of manufacturing processes, real-time condition monitoring (at all levels and in all powertrain components), energy management optimization, repurposing of the components, and even recycling processes. This paper presents an overview of different DT platforms that can be used in EV applications. A deductive comparison between model-based and data-driven DT was performed. EV main systems have been discussed regarding the usable DT platform. DT platforms used in the EV industry were addressed. Finally, the review showed the superiority of data-driven DTs over model-based DTs due to their ability to handle systems with high complexity.

Keywords: digital twin; electric vehicle; platform; software

1. Introduction

In recent decades, digital manufacturing has contributed significantly to all industries. The remarkable advances in communication and information technology have gone a long way towards the development of manufacturing [1]. Computer-aided technologies such as computer-aided design (CAD), computer-aided engineering (CAE), computer-aided manufacturing (CAM), finite element analysis (FEA), product data management (PDM), etc., are developing rapidly and play a crucial role in modern industry [2].

Manufacturing, healthcare, and smart city environments have become more able to harness data through advanced analytics and the Internet of Things (IoT) connectivity [3]. In conjunction with data analytics, IoT environments can be used for predictive maintenance, fault detection, and design optimization processes [4]. When it comes to describing, finding, and accessing resources, DTs and IoT overlap. DT and IoT standards have been developed by many organizations with various backgrounds and perspectives to address these overlapping aspects. IoT and DT both focus on resources [5]. Resources are internet-connected objects that can communicate with consumers either directly or indirectly through some sort of software system in the context of the IoT. Resources are defined more broadly in the context of DT, including assets, devices, and actual or virtual entities. Both share the concept that most resource-to-resource communication, or machine-to-machine (M2M) communication, should occur without the involvement of humans. With the advancements in DT technology, the gap between IoT and data analytics can be bridged by creating connected physical and virtual models [6]. This has allowed DT technology to be applied in many different sectors and disciplines such as smart cities, construction, healthcare, ocean, automobile, aerospace, manufacturing, utilities, etc. [7].

Citation: Ibrahim, M.; Rjabtšikov, V.; Gilbert, R. Overview of Digital Twin Platforms for EV Applications. *Sensors* **2023**, *23*, 1414. https://doi.org/10.3390/s23031414

Academic Editor: Giambattista Gruosso

Received: 22 November 2022
Revised: 9 January 2023
Accepted: 24 January 2023
Published: 27 January 2023

1.1. Background

After Challenge Advisory hosted Michael Grieves' presentation on technology at the University of Michigan in 2002, the concept of the DT gained wider recognition [8]. During this presentation, the focus was on the development of a lifecycle management center for products. The presentation covered all the key details associated with DT technology, such as the physical and digital environment, as well as the transfer of appropriate information and data between the physical and digital worlds. The DT concept has been practiced since the 1960s by NASA during the space programming period. They created physically duplicated systems at ground level to match the systems in space [9].

The term DT refers to the digital representation of a physical object, process, or service that supports decision-making throughout its lifecycle. It is updated from real-time data and uses simulation, machine learning, and reasoning [10]. With improved data accessibility and connection and the changing end-user needs, the idea of DT can be considered a logical extension of conventional simulations [1]. It is a computer program that simulates how products and processes will perform using real-world data. Software analytics, artificial intelligence, and the Internet of Things can be integrated into these programs to enhance the output. Three basic pillars make up the DT, which are the physical entity, the virtual entity, and the data exchange and communication system between them [11]. Creating a DT for a system is a multiphase process comprises of modeling, validation, training, and deployment [12].

Recent works have defined DT technology as a five-dimensional structure with separate entities for services and connections [8]. Creating a DT can enhance technology trends, prevent costly failures in physical objects, and improve test processes by using advanced analytics, monitoring, and predictive capabilities. Figure 1 shows the main structure of DT.

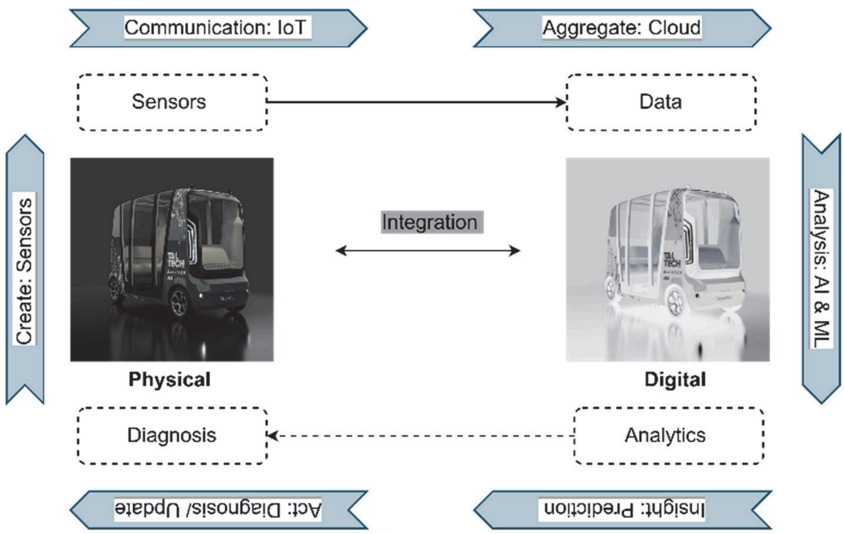


Figure 1. Systematic Characteristics of DT.

1.2. Digital Twin in EV Industry

Historically, automotive and aerospace systems have been developed using empirical engineering practices [13], but now with growing performance requirements, the necessity for "self-awareness" during operation, and the necessity for a lack of external assistance, new engineering procedures are required. With the emergence of DT, new testing and development modeling techniques have become available to fulfil new requirements. As a result, research interest in these technologies had also increased steadily, as illustrated in Figure 2.

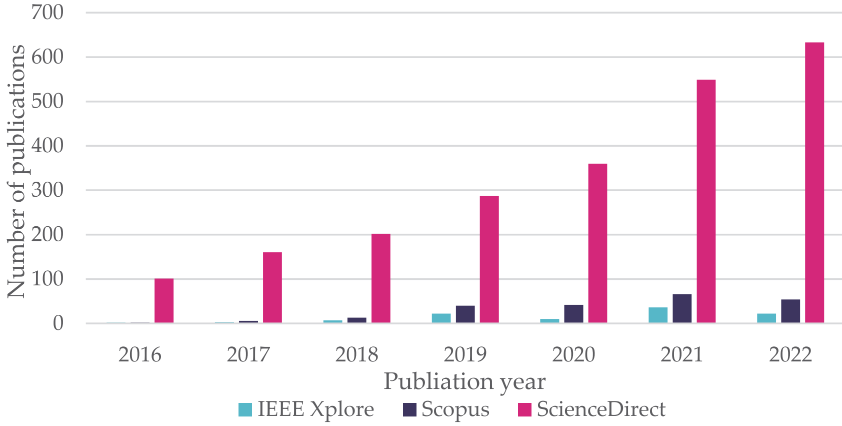


Figure 2. Search results for publications related to DT in automotive applications during the period 2011–2022 in ScienceDirect and Scopus.

The EV industry is gaining increased attention nowadays. The rising demand for EVs is because they not only eliminate exhaust emissions and contribute to the transportation sectors (23% of global CO_2 emissions), but because they also provide critical grid flexibility as a transition to a greater share of renewable energy (RE) supply. Despite this solid strategy, EVs accounted for only 7.2% of global car sales in 2021. Pricing and battery capacity pose major challenges to the introduction of EVs on the road. To address these challenges, one way is to optimize the electrical energy consumption of the vehicle and design a supporting architecture to facilitate it. As the 4th industrial revolution presses on, EV manufacturers are adopting even more technology to make their production operations proceed and make them more cost-effective. Advanced machine learning tools and optimization algorithms have contributed highly to the EV development process [13]. The IoT, along with DT, act as the required architecture for mapping offline physical assets to digital models. Since EVs generate significant amounts of sensory data, the DT technology is far superior to other technologies such as hardware-in-the-loop (HIL) simulations. Smart system monitoring, predictive events, fault detection, remaining useful lifespan, and many other benefits can be achieved through this conversion. Despite the many advantages that DT offers to the technology of manufacturing and developing EVs, mastering this application is still in the early stages. EVs comprise a mixture of electrical and mechanical systems that range in complexity. One of the main problems facing researchers in this regard lies in choosing an appropriate development environment (platform) to create a DT of an EV system.

This paper presents a comprehensive overview of different platforms used to develop DTs for EV applications. The general objective of this study is to provide a reference for researchers on this topic. The paper is organized as follows: A systematic understanding of the inception and evolution of DT technology and its implementation in automotive applications is offered in Section 1. Section 2 highlights and compares the two main categories of DTs. In Section 3, the study investigates DT platforms for potential contributions to EV technologies and considers current barriers to their realization. Section 4 addresses the research findings for innovation in this field. Finally, Section 5 concludes the main findings and presents recommendations for future work.

2. DT Architecture Categorizations

The DT architecture can be divided into two main categories as the following subsections illustrate.

2.1. Model-Based DT

The concept of a model-based simulation approach (MBS) refers to a formalized methodology for preparing requirements and designing, analyzing, and verifying complex systems [14]. MBS places models at the center of the system design. Physical systems, whether in nature, on the testbench, or in applications, consist of interconnected and interacting objects or components performing a task or a variety of functions. Simulating a physical system using MBS implies that the mechanism inside the system is being studied using fundamental physical laws and principles of engineering. The power of MBS relies on a deep understanding of the system or process and can benefit from scientifically established relationships. Model-based DT is an advanced form of MBS with increased sensory data and AI supplementary tools. The following literature illustrates some examples of model-based DTs and the used platforms for creation in different applications. Madni et al. [15] implemented DT technology in a model-based system of a vehicle using a planar mechanics open-source library. Bachelor et al. [16] proposed a case study of a model-based DT of an ice protection system for a regional aircraft using Dassault Systems' Dymola platform. Magnanini and Tullio [17] proposed an analytical model-based DT of a railway axles manufacturing system for a performance evaluation based on Markovian system representation. Zheng and Sivabalan [18] used a Windows Presentation Foundation (WPF) application and .Net framework 4.5 in Visual Studio to develop a DT for a cyber-physical system (CPS) of a 3D printer based on a tri-model-based approach for product-level development. Ward et al. [19] proposed a model-based machining DT system for a case study of a large-scale CNC machine tool using a MATLAB/Simulink platform. Yang et al. [20] developed a model-based DT of an aero-engine disk for online detection of disk unbalance and crack failure using an ANSYS simulation platform. Woitsch et al. [21] proposed a meta-model of a model-based DT environment to bridge the between the manufacturing and the use of products and services based on an ADOxx meta-modeling platform.

From the above, it is clear that the creation of a model-based DT of a system is closely related to modellable physical systems and mostly depends on conventional modeling and simulation platforms, in addition to some artificial intelligence techniques and IoT tools. Although model-based DTs are widely used in different applications, some obstacles undermine their use, especially with high-complexity systems. The major drawback of model-based DT is that models cannot handle infinite complexity and typically need to be simplified. Moreover, it has difficulty considering unknown variables and noisy data.

2.2. Data Driven DT

The adoption of DTs enables operators to monitor production, test deviations in an isolated virtual environment, and strengthen the security of process industries [6]. With the substantial increase in process data, conventional model-based methods are unable to describe complex systems' state space. In this way, data-driven modeling technology has become a potential solution for modeling DTs. The data-driven modeling concept is based on analyzing data about a system to find connections between variables (input, internal, and output variables) without explicitly knowing its physical behavior. As compared to conventional empirical models, these methods represent a significant advance in a wide range of applications. Data-driven modeling relies on substantial and sufficient data to describe the underlying system. Data are used to perform tasks such as classification, pattern recognition, associative analysis, and predictive analytics. The literature shows excessive use of data-driven DT in different applications especially systems with high complexity as will be described in the following. Wang et al. [22] developed a data-driven DT framework for a three-domain mobility system of human, vehicles, and traffic based on an Amazon web services (AWS) platform. Gao et al. [23] used a MATLAB/Simulink platform to build an anomaly detection framework for monitoring anomalous behaviors in a data-driven DT-based cyber-physical system. Coraddu et al. [24] developed a data-driven DT of a ship for speed loss and marine fouling estimation based on a large number of onboard sensors using the IBM Engineering Lifecycle Management (IBM-ELM) platform.

Merghani et al. [25] proposed a data-driven DT of a proton exchange membrane fuel cell (PEMFC) for system health monitoring and lifetime prediction. Mykoniatis and Harris [26] implemented a data-driven DT of an automated mechatronic modular production system for condition monitoring, design decisions, testing, and validating the actual system behavior using the Any Logic Simulation platform. Blume et al. [27] developed a data-driven DT of a cooling tower for improving system understanding and performance prediction using the software tools KNIME and Microsoft Excel. Kim et al. [28] developed a data-driven DT of an onload tap charger (OLTC) for health monitoring and fault detection based on a numerical algorithm of subspace state-space system identification (N4SID). Major et al. [29] developed a java-based data-driven 3D graphical DT platform for smart cities applications. They also supported their study with a real study case of a smart city in Norway.

From the foregoing, it is obvious that there is a direct connection between the data-driven DT and the complex systems that contain a huge amount of data. It is also noted that the platforms used for data-driven DT creation are often artificial intelligence and Big Data tools. Table 1 summarizes the comparison between data-driven and model-based DTs.

Table 1. Comparison between model-based and data-driven DTs from different perspectives.

Comparison	Model-Based DT	Data-Driven DT
Basis	Mathematical equations of physical lows (Model Simulation)	Sensory data of system's inputs and outputs (grayor black box)
Cost	More expensive	Less expensive
Time of creation	Shorter	Longer
Applications	Modellable physical systems	Cyber-physical systems, complex systems

3. DT Platforms for EV Applications

EVs are also referred to as battery electric vehicles (BEV), as they use a battery pack to store the electrical energy that powers the electric motor. EV main domains are divided into two categories as follows: a smart vehicle system and a vehicle propulsion drive system.

3.1. Smart Vehicle System

Emerging technologies in the field of smart vehicle systems have promoted the continuous development of sustainable transport. To increase energy efficiency and reduce CO_2 emissions, smart electric vehicles have been deployed to achieve decarbonization challenges. The smart vehicle system includes advanced driver assistance systems and vehicle health management systems. Bhatti et al. [30] conducted research to provide a comprehensive analysis of DT for smart electric vehicle applications, which highlighted the implementation of DT platforms for health monitoring systems based on integrated vehicle health management (IVHM).

Sanabria et al. [31] developed a DT of an electric passenger bus to emulate the vehicle's performance. They provided predictive maintenance models to determine the remaining useful life of the vehicle components. They used the MATLAB/Simulink platform deployed on an NVIDIA processor through Compute Unified Architecture (CUDA).

Ezhilarasu et al. [32] discussed the prospective role of DT in an integrated vehicle health management system (IVHMS) to support condition-based maintenance (CBM) by monitoring, diagnosing, and prognosing the vehicle health.

Advanced driver assistance systems are also a point of interest not only for increasing energy savings but also for achieving a more comfortable driving experience. Sun et al. [33] used MATLAB Simulink and Carsim to deploy machine learning algorithms and developed a more accurate and precise groundwork for training and testing smart vehicle DTs.

Wang et al. [34] developed a DT of an advanced driver assistance system for a connected and automated vehicle (CAV) by leveraging the Unity game engine as a physical system emulator. They built the DT virtual model using e Unity scripting API combined with external tools (e.g., SUMO, MATLAB, Python, and/or AWS) to enhance the simulation functionalities. To provide reliable and safe online monitoring for autonomous guided vehicles (AGVs), El Sisi et al. [35] integrated an IoT architecture to address the issue of cyber-attacks based on a deep neural network (DNN) with a rectified linear unit.

Lui et al. [36] proposed two approaches based on a Gaussian process (GP) and a deep convolutional neural network (DCNN) for DT model development of a heavy vehicle for optimization of vehicle driving states.

The advantages of DT technologies integrate autonomous navigation performance; however, critical decision-making must be considered to enable the modelling of large vehicle data. Bottani et al. [37] developed a DT for preparing the AGV control system using discrete event simulation software (DES) based on the Arduino and C++ interpreter.

The ability to introduce several scenarios for critical decision-making provides a more accurate model through the application of stochastic factors using a DT platform; therefore, physical assessment is also required. Guerra et al. [38] proposed the optimization of a DT for modeling the behavior of ultraprecision motion systems with backlash and friction. The implementation of the complete algorithm and simulation was performed using MATLAB/Simulink, concluding that the cross-entropy method required a remarkably shorter time compared to other optimization approaches; hence, further studies are necessary to analyze the influence of different optimization methods.

3.2. EV Propulsion Drive System

The EV powertrain is the main system that defines a vehicle as an EV. It is a combination of electrical and mechanical components. Figure 3 shows different components of an EV propulsion drive system.

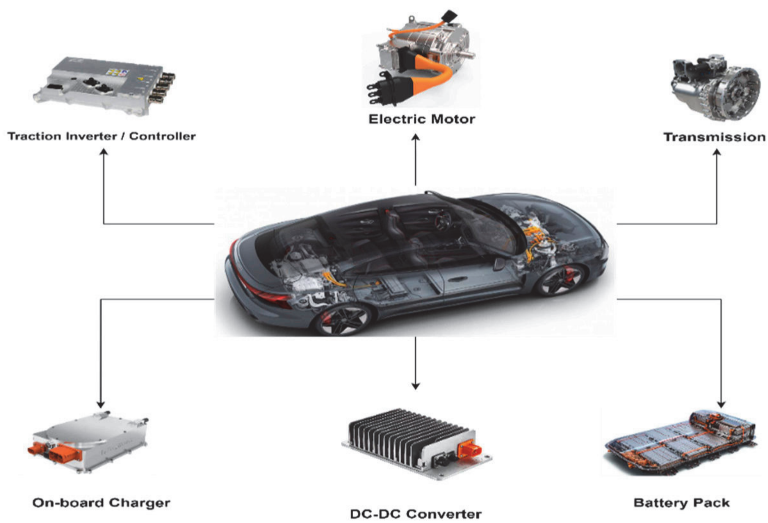


Figure 3. Main components of an EV propulsion drive system.

Despite the multiple components in the electric propulsion system, most research efforts in EV digital twin technology are focused on three specific components: the battery, the electric motor, and the traction inverter/controller.

3.2.1. EV Battery System

Digital twin applications for a battery energy storage system (BESS) is an important topic that contributes to sustainability and climate change mitigation, not only by reducing CO_2 emissions but also by implementing green strategies towards clean energy sources.

The battery management system (BMS) is defined as the core element of a battery that monitors, protects, and ensures reliability, safety, and efficiency [39]. It is fundamental to point out that some indicators play a fundamental role in the successful BESS implementation, such as the state of charge (SOC), state of health (SOH), depth of charge (DOC), and depth of discharge (DOD).

Several scientific studies have been conducted to determine the major relevant applications of DTs for battery systems. In 2020, Wu et al. [40] used Python Battery Mathematical Modelling (PyBaMM) and MATLAB to propose the introduction of hybrid models, defined as models that combine physics-based models and data-driven approaches. Wu et al. also mention opportunity areas in the fields of (1) standardized and transparent data, (2) a combination of machine learning and artificial intelligence algorithms, and (3) development of new methodologies to assess lifetime assessment of battery systems [41].

Concerning health and charge indicators, a cloud BMS was implemented by using software programs in Python, in which cloud computing was used to improve computational power data as well as storage capacity. The research contribution proposed by Li et al. is explained in the following points [42]:

- SOC and SOH estimations to validate particle swarm optimization: In this case, aging tests were carried out for both software and hardware. Additionally, a battery test for lead-acid and lithium-ion batteries was performed to validate the results of SOC and SOH estimations;
- Battery Modeling: Implementation of the equivalent circuit model (ECM) was executed with additional modifications to the battery dynamics, taking into consideration the particle swarm optimization (PSO) and the adaptative extended H-infinity filter (AEHF);
- Cloud BMS: A DT was built to improve the computation power, data storage capability of a BMS, and reliability, all this considering the concept of IoT and cloud computing.

Future research that identifies the efforts to implement a BESS for DT was also proposed by Singh et al. in 2021, highlighting software packages in Python and MATLAB. The most important benefits of the DT and the integrated BMS in the scientific study conducted by Singh et al. were the following [39]: (1) evaluation of the battery performance, (2) aging indicators to predict the remaining useful lifetime (RUL), (3) optimal assessment of the SOC, (4) thermal management, and (5) fault diagnostics.

Selection of an optimal algorithm before building the DT is a challenging task to accomplish, all due to the specifications of battery packs, input data, operating conditions, and manufacturing requirements that a BESS must fulfil. Sancarlos et al. [43] developed a regression model based on sparse-proper generalized decomposition (s-PGD) that was incorporated into a DT, allowing for not only real-time simulation but also to achieve battery evaluation and early prediction (BEEP). It is important to mention that a data-driven model was also implemented to provide more optimal accuracy that corrects the results between the prediction and measurements. Finally, it was summarized that improvements to the DT model can be incorporated by considering not only thermal gradient but also aging effects as a future line of research. Results and validation models were compared using lithium-ion simulation battery toolbox (LIONSIMBA) in MATLAB.

Regarding the analysis of degradation mechanisms in BESS, points of interest are sustained in the aging and RUL of the system. Operating temperatures are the major indicator of heat generation in the battery pack. Soleymani et al. [44] generated a semi-analytical DT model to capture thermal behavior in a real-time environment. The proposed model was used to accelerate the battery pack design and development through the evaluation of several operating conditions such as charge and discharge profiles, initial SOC, coolant flow rate, and temperature. Results of the research were illustrated in ANSYS and provide an optimization for reliability, comfort, and safety in battery pack thermal systems, which results in a significant reduction in time-to-market.

To conclude with this section, it is necessary to point out that the major requirements of the DT implementation in a BESS are based on a solid understanding of the physical system, selection of the most optimal model based on input data and manufacturing requirements, execution of the data-driven approach according to the key performance indicators (KPIs), and finally, assessing the fault diagnostics and predictive maintenance by testing processes and BMS specifications.

The continuous advance in the IoT has encouraged the development of new software platforms for battery data storage; all this ensures easy access by the creation of learning models that guide the product design and optimization processes. In [45], battery data storage platforms simplify the prediction of the RUL, which supports not only the design usage history, but also the behavioral integration in consequent life cycle phases. It is important to mention that the big data platforms must fulfil the performance of integration, storage, interactive analysis, visualization, and security, all to assure the implementation of advanced technological tools, such as sensor data, model generation data, multiple structures, real fusion, and virtual data.

Execution and deployment of software platforms for implementing the DT of a BESS is a fundamental step that can be summarized in the next points [39]:

- Use of experimental inputs to determine parameter identification.
- Implementation of the state estimation algorithm.
- Integration of a battery modeling that considers the design and manufacturing data.
- Execution of the parameter-update estimation that can be coded in several tools, such as MATLAB, Python, Linux, etc.

The variety in existing libraries and open-source battery modeling based on software packages is the most crucial step for results delivery. Although the selection of the software package depends on the sector, it has been proven by scientific studies that MATLAB, COMSOL, Dualfoil, and fast DFN have improved the performance and functionalities of the models, not only in the academic field but also for industrial purposes.

Considering the parameter estimation, the PyBaMM platform is considered a powerful tool to facilitate computational complexity by solving standard electrochemical battery models [46]. The feasibility of PyBaMM execution and its main contributions relies on the following advantages and customized attributes [39]: (1) boundary conditions in the initialization of the algorithm, (2) governing equations based on electrochemical models, (3) initial conditions, (4) output variables of the model that represent the internal state of the battery, and (5) customized attributes that illustrate the physical meaning of the system (termination events, battery region, geometry, and computation solver).

Special DT platforms have also been implemented to assess the performance degradation of lithium-ion batteries. Peng et al. [47] developed a low-cost DT based on LabView 2018 using an equivalent circuit model (ECM) to realize a pack degradation assessment of lithium-ion battery packs. Among their main contributions was a DT platform to test different battery types and load algorithms for SOC estimation. Their results indicated that their platform provides accurate new solutions for battery degradation in real-time; however, compatibility with different algorithms and incorporation of new features, such as virtual reality and augmented reality, are opportune areas for further improvement.

In terms of challenges regarding data and sensing of standardized collection methods, numerous efforts have been proposed to achieve suitable data structures and effective data-driven approaches. One remarkable effort was developed by Herring et al. [48], a scientific study in which a BEEP Python library was implemented, enabling cell-testing and machine-learning applications.

3.2.2. EV Electric Motor

The electric motor is considered the core element of an EV. It is responsible for converting electric energy from the battery into kinetic energy that moves the vehicles' wheels. It functions in part as an electric generator, converting kinetic energy created when the vehicle is in neutral (for example, when it is descending a slope) into electric energy that is stored in the battery. When the car decelerates, the same energy-saving concept is used, resulting in a "regenerative braking system". The main challenges of EV motors concern their design and control [49]. The main goal is achieving maximum efficiency of the motor, which means higher driving range and longer battery life [50]. The advancement in DT technology has coped with many problems of motor design and control. DT technology provides many advantages for EV motors, from design optimization to prognosis and determining the life span of different parts. In the meantime, DT technology facilitated motor control algorithm development. The control strategy can be implemented and tested through the motor DT without the need for a real physical model, which saves a lot of time and power consumption needed for testbench development. Many platforms for electric machine design and control support DT creation and deployment as shown in the literature.

Venkatesan et al. [51] proposed an intelligent DT model of an EV PMSM for health monitoring and prognosis. The MATLAB/Simulink platform supported with an artificial neural network (ANN) and Fuzzy logic tools were used to build the motor DT. Rassolkin et al. [52] used MATLAB/Simulink and Unity 3D platforms to build a DT of an induction motor for condition monitoring. Goraj [53] used Siemens' product lifecycle management (PLM) platform to build a DT of an airplane electric motor for lifetime fatigue prediction analysis. Proksh et al. [54] developed an empirical-based DT model of an induction motor using MATLAB/Simulink to monitor the bearing voltage and electric breakthroughs. Jitong et al. [55] used 3D MAX and Unity 3D platforms to build a DT of a three-phase induction motor for condition monitoring of motor equipment. Ruba et al. [56] presented a DT for an EV propulsion system based on energetic macroscopic representation (EMR) using the LabVIEW platform. Abbate et al. [57] developed a DT approach for an industrial electric motor to evaluate its behavior based on vibration data for maintenance purposes using the Arena simulation platform. Bouzid et al. [58] proposed a real-time DT of a wound rotor induction motor for condition monitoring based on FEM of the motor using RT-LAB in the MATLAB/Simulink environment. Ibrahim et al. [59] proposed a DT of an EV-PMSM based on the motor analytical model to act as a virtual torque sensor. They used the MATLAB/Simulink platform combined with the Robot Operating System (ROS) to build the motor DT.

3.2.3. Traction Inverter

Power electronics interfaces are a key element in enabling the transition from conventional internal combustion engine vehicles (ICV) to EVs [60]. Traction inverter technology has recently advanced, making it a particularly promising field for expansion. The traction inverter controls how much energy is transferred from the high-voltage battery system to the motor, which turns the wheels and moves the vehicle. Inverters contain motor control units (MCU), which are usually integrated parts. The EV motor's control algorithm is implemented by the MCU. As soon as it receives comments from the vehicle control unit (VCU) via CAN-bus communication, it configures motor speed and torque, which are then converted by the inverter into power signals. An inverter is considered the brain of the EV as it is the main link between stationary and kinetic elements. Insulated gate bipolar transistors (IGBT) have been the base element of EV inverters since 1980. Field-effect transistors (FETs) with simple gate-drive and bipolar transistors (BJTs) with high current and low conduction loss were merged to create IGBT. With low on-state conduction losses, as well as a strictly controlled switching rate, IGBTs can block high voltages. Despite their fast-switching capabilities, they suffer from low on-state conduction losses. As a result, they require a larger thermal management system which has a negative impact on the power conversion system efficiency.

Power transistors made of silicon carbide (SiC) and gallium nitride (GaN) have recently gained popularity as IGBT substitutes. [61]. By switching at higher frequencies (100 kHz or more as opposed to 20 kHz), SiC devices can increase efficiency while minimizing the size and cost of any inductors or transformers [62]. GaN transistors have been used in a range of switch-mode power supply applications, including DC/DC converters, inverters, and battery chargers because of their ability to tolerate high voltages (up to 1000 V), high temperatures, and fast switching [63]. The main drawback of such a technology is that it is still high costs.

The advancement in DT technology for EV inverters has had a significant effect. Health monitoring, fault diagnosis, performance optimization, and lifetime estimation of semiconductors are the main prospective functions of DT for EV inverters as the literature shows. Milton et al. [64] proposed a DT of a power converter running on a field programmable gate array (FPGA) for online diagnostic analysis using the MATLAB platform. Wunderlich and Santi [65] developed a data-driven DT model of a power electronic converter based on a dynamic neural network for condition monitoring using the MATLAB platform. Liu et al. [66] proposed a model-based DT of a power electronic converter for condition monitoring using the MATLAB/Simulink platform. Wu et al. [67] proposed a DT approach for a single-phase inverter for degradation parameters identification using the MATLAB platform. Shi et al. [68] proposed a DT method for IGBT parameter identification of a three-phase DC/AC inverter for circuit condition motoring based on a particle swarm optimization algorithm using the MATLAB/ Simulink platform. Liu et al. [69] developed and experimentally validated a DT of an automotive traction drive system. The proposed DT combined an FEM-based PMSM model with a SiC inverter circuit simulation using the MATLAB /Simulink platform.

3.3. DT Platforms from EV Industry

Many producers of EVs and their co-systems are using the DT platform for research and development purposes. Some EV manufacturers have established their own DT platforms, while others are in collaboration with global platform developers [70–73]. Table 2 provides an adequate review of some DT platforms used by EV manufacturers.

Table 2. Some DT platforms of EV manufacturers and their functions.

Manufacturer	DT Platform	Origin	Function
BMW	Nvidia Omniverse	Nvidia	Predictive maintenance, Virtual factory planning, Condition monitoring
General Electric	Smart Signal	General Electric	Condition monitoring, Fault detection, Diagnosis, Forecasting
Hyundai	Azure	Microsoft	Predicting EV battery lifespan, optimizing battery management and performance
Kia	NX software	Siemens	Design optimization, Predictive maintenance
Siemens	Siemens Xcelerator	Siemens	Testing simulations and calculations on digital versions
Bosch	Bosch IoT Suite	Bosch	Condition Monitoring, Product testing
Mitsubishi	MELSOFT Gemini	Mitsubishi	Visualization, Design optimization, Predictive maintenance
Skoda Auto	Matterport DT	Matterport	Condition monitoring

4. Discussion

The first key step of creating a DT for a system is modeling. It is necessary to choose between the two main DT modeling architectures: data-driven and model-based. The selection relies on several factors, including the function performed by the DT, the system parameter availability, and the simplicity or the complexity of the system. The next step of the DT development process is to choose the right development environment (platform).

From the perspective of EV applications, EV vehicles were divided into two main domains: the smart vehicle system and the vehicle propulsion drive system. Creating a DT of a smart vehicle system is more achievable based on data-driven techniques. Whilst for EV propulsion systems, a mixture of data-driven, model-based, or hybrid DT architectures have been applied. For battery storage systems, including battery health management systems, data-driven DTs showed more reliability and flexibility; however, some researchers used a hybrid architecture to model the system. In contrast, electric motors and traction inverters can be modeled in diverse ways such as by finite element (FEM), analytical, and numerical models; thus, they were modeled more by model-based DTs.

The use of platforms such as MATLAB/Simulink, Ansys, LabView, Unity 3D, and other modeling platforms has been effective in creating model-based DTs. While in the case of data-driven DTs, more reliance has been on cloud-based platforms such as Microsoft Azure, AWS, IBM-ELM, or special purposes platforms built by the DT developers based on one of the software development environments, such as Python, C++, R, and others. Figure 4 represents an illustrative figure summarizing the results of this review of DT architectures for different EV systems.

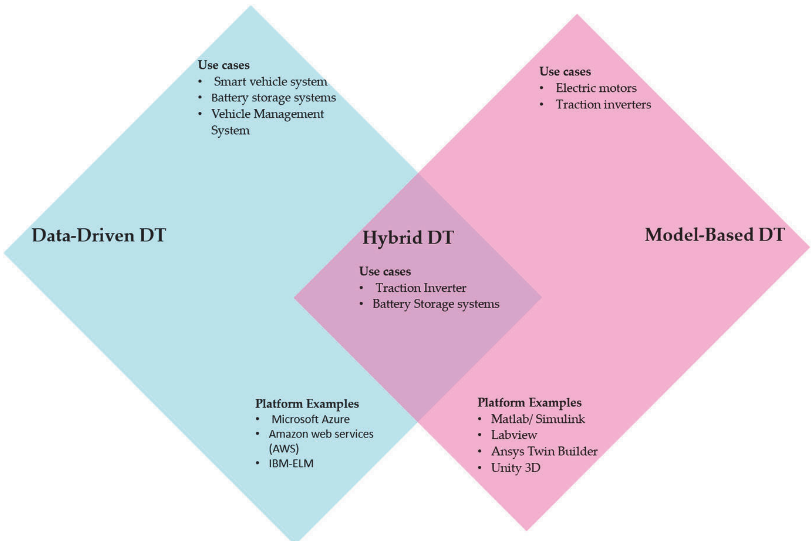


Figure 4. DT architectures for different EV systems.

5. Conclusions

Recently, DTs have become an emerging paradigm for virtual representations of complex systems along with their underlying components.

DTs are composed of three main parts: physical objects, virtual representations, and the communications between them. The virtual part of DT must be developed through a specific environment called the DT platform.

Model-based and data-driven are the main categories of DTs. A comparison between the two categories clarified their strengths and weaknesses as well as the prospective applications for both.

This review dealt specifically with DTs for EV applications. EV systems were divided into smart vehicle systems and vehicle propulsion drive systems. The literature addressed the advantages of using data-driven DTs with smart vehicle systems due to the complexity of modeling such systems and also the significant amount of data concerned with it. While in the case of the electric propulsion drive system, there was mixture between the use of model-based DT, data driven DT, or a combination of them both, depending on the component to be modeled and the DT's function.

This paper represents a reference for researchers on the topic of DT for EV applications in order to determine the appropriate DT platform according to the work requirements.

For researchers, many platforms may be used to create DTs for different EV systems, but the reality in industry may differ slightly. Most EV manufacturers rely on their unique platforms for research and development purposes. The main issue with such platforms is that they are not open source, which deepens the gap between academic research and industrial development.

Author Contributions: Conceptualization, M.I. and V.R.; methodology, M.I.; validation., M.I. and R.G.; formal analysis, M.I.; investigation, R.G.; resources, M.I.; data curation, V.R.; writing—original draft preparation, M.I.; writing—review and editing, M.I.; visualization, V.R.; supervision, M.I.; project administration, M.I.; funding acquisition, M.I. All authors have read and agreed to the published version of the manuscript.

Funding: The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of an autonomous electric vehicle".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ibrahim, M.; Rassõlkin, A.; Vaimann, T.; Kallaste, A. Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System. *Sustainability* **2022**, *14*, 601. [CrossRef]
2. San, O. The digital twin revolution. *Nat. Comput. Sci.* **2021**, *1*, 307–308. [CrossRef]
3. Singh, M.; Fuenmayor, E.; Hinchy, E.P.; Qiao, Y.; Murray, N.; Devine, D. Digital Twin: Origin to Future. *Appl. Syst. Innov.* **2021**, *4*, 36. [CrossRef]
4. Boje, C.; Guerriero, A.; Kubicki, S.; Rezgui, Y. Towards a semantic Construction Digital Twin: Directions for future research. *Autom. Constr.* **2020**, *114*, 103179. [CrossRef]
5. Jacoby, M.; Usländer, T. Digital Twin and Internet of Things—Current Standards Landscape. *Appl. Sci.* **2020**, *10*, 6519. [CrossRef]
6. Jazdi, N.; Talkhestani, B.A.; Maschler, B.; Weyrich, M. Realization of AI-enhanced industrial automation systems using intelligent Digital Twins. *Procedia CIRP* **2021**, *97*, 396–400. [CrossRef]
7. Wu, J.; Yang, Y.; Cheng, X.U.N.; Zuo, H.; Cheng, Z. The Development of Digital Twin Technology Review. In Proceedings of the 2020 Chinese Automation Congress (CAC 2020), Shanghai, China, 6–8 November 2020; pp. 4901–4906. [CrossRef]
8. Tao, F.; Zhang, M.; Nee, A.Y.C. Five-Dimension Digital Twin Modeling and Its Key Technologies. In *Digital Twin Driven Smart Manufacturing*; Academic Press: Cambridge, MA, USA, 2019; pp. 63–81.
9. Glaessgen, E.H.; Stargel, D.S. The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. In Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA, Honolulu, HI, USA, 23–26 April 2012.
10. Rassõlkin, A.; Orosz, T.; Demidova, G.L.; Kuts, V.; Rjabtšikov, V.; Vaimann, T.; Kallaste, A. Implementation of digital twins for electrical energy conversion systems in selected case studies. *Proc. Est. Acad. Sci.* **2021**, *70*, 19–39. [CrossRef]
11. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [CrossRef]
12. Liu, M.; Fang, S.; Dong, H.; Xu, C. Review of digital twin about concepts, technologies, and industrial applications. *J. Manuf. Syst.* **2021**, *58*, 346–361. [CrossRef]
13. Eppinger, S.D.; Whitney, D.E.; Smith, R.P.; Gebala, D.A. A model-based method for organizing tasks in product development. *Res. Eng. Des.* **1994**, *61*, 1–13. [CrossRef]
14. Elsisi, M. Improved grey wolf optimizer based on opposition and quasi learning approaches for optimization: Case study autonomous vehicle including vision system. *Artif. Intell. Rev.* **2022**, *55*, 5597–5620. [CrossRef]
15. Madni, A.M.; Madni, C.C.; Lucero, S.D. Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems* **2019**, *7*, 7. [CrossRef]
16. Bachelor, G.; Brusa, E.; Ferretto, D.; Mitschke, A. Model-Based Design of Complex Aeronautical Systems through Digital Twin and Thread Concepts. *IEEE Syst. J.* **2020**, *14*, 1568–1579. [CrossRef]
17. Magnanini, M.C.; Tolio, T.A.M. A model-based Digital Twin to support responsive manufacturing systems. *CIRP Ann.* **2021**, *70*, 353–356. [CrossRef]
18. Zheng, P.; Sivabalan, A.S. A generic tri-model-based approach for product-level digital twin development in a smart manufacturing environment. *Robot. Comput. Integr. Manuf.* **2020**, *64*, 101958. [CrossRef]

19. Ward, R.; Sun, C.; Dominguez-Caballero, J.; Ojo, S.; Ayvar-Soberanis, S.; Curtis, D.; Ozturk, E. Machining Digital Twin using real-time model-based simulations and lookahead function for closed loop machining control. *Int. J. Adv. Manuf. Technol.* **2021**, *117*, 3615–3629. [CrossRef]
20. Yang, Y.; Ma, M.; Zhou, Z.; Sun, C.; Yan, R. Dynamic Model-based Digital Twin for Crack Detection of Aeroengine Disk. In Proceedings of the 2021 International Conference on Sensing, Measurement & Data Analytics in the era of Artificial Intelligence (ICSMD), Nanjing, China, 21–23 October 2021. [CrossRef]
21. Woitsch, R.; Sumereder, A.; Falcioni, D. Model-based data integration along the product & service life cycle supported by digital twinning. *Comput. Ind.* **2022**, *140*, 103648. [CrossRef]
22. Wang, Z.; Gupta, R.; Han, K.; Ganlath, A.; Ammar, N.; Tiwari, P. Mobility Digital Twin with Connected Vehicles and Cloud Computing. 2022. Available online: https://www.techrxiv.org/articles/preprint/Mobility_Digital_Twin_with_Connected_Vehicles_and_Cloud_Computing/16828759/1 (accessed on 22 November 2022). [CrossRef]
23. Gao, C.; Park, H.; Easwaran, A. An anomaly detection framework for digital twin driven cyber-physical systems. In Proceedings of the ACM/IEEE 12th International Conference on Cyber-Physical Systems 11, Nashville, TN, USA, 19–21 May 2021; pp. 44–54. [CrossRef]
24. Coraddu, A.; Oneto, L.; Baldi, F.; Cipollini, F.; Atlar, M.; Savio, S. Data-driven ship digital twin for estimating the speed loss caused by the marine fouling. *Ocean Eng.* **2019**, *186*, 106063. [CrossRef]
25. Meraghni, S.; Terrissa, L.S.; Yue, M.; Ma, J.; Jemei, S.; Zerhouni, N. A data-driven digital-twin prognostics method for proton exchange membrane fuel cell remaining useful life prediction. *Int. J. Hydrogen Energy* **2021**, *46*, 2555–2564. [CrossRef]
26. Mykoniatis, K.; Harris, G.A. A digital twin emulator of a modular production system using a data-driven hybrid modeling and simulation approach. *J. Intell. Manuf.* **2021**, *32*, 1899–1911. [CrossRef]
27. Blume, C.; Blume, S.; Thiede, S.; Herrmann, C. Data-Driven Digital Twins for Technical Building Services Operation in Factories: A Cooling Tower Case Study. *J. Manuf. Mater. Process.* **2020**, *4*, 97. [CrossRef]
28. Kim, W.; Kim, S.; Jeong, J.; Kim, H.; Lee, H.; Youn, B.D. Digital twin approach for on-load tap changers using data-driven dynamic model updating and optimization-based operating condition estimation. *Mech. Syst. Signal Process.* **2022**, *181*, 109471. [CrossRef]
29. Major, P.; Li, G.; Hildre, H.P.; Zhang, H. The use of a data-driven digital twin of a smart city: A case study of ålesund, norway. *IEEE Instrum. Meas. Mag.* **2021**, *24*, 39–49. [CrossRef]
30. Bhatti, G.; Mohan, H.; Singh, R.R. Towards the future of smart electric vehicles: Digital twin technology. *Renew. Sustain. Energy Rev.* **2021**, *141*, 110801. [CrossRef]
31. Botín-Sanabria, D.M.; Santiesteban-Pozas, D.A.; Sáenz-González, G.; Ramírez-Mendoza, R.A.; Ramírez-Moreno, M.A.; Lozoya-Santos, J.D.J. Digital Twin for a Vehicle: ElectroBus Case Study. In Proceedings of the International Conference on Industrial Engineering and Operations Management, Monterrey, Mexico, 3–5 November 2021.
32. Ezhilarasu, C.M.; Skaf, Z.; Jennions, I.K. Understanding the role of a digital twin in integrated vehicle health management (IVHM). In Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 6–9 October 2019; pp. 1484–1491. [CrossRef]
33. Sun, B.; Deng, W.; He, R.; Wu, J.; Li, Y. Personalized Eco-Driving for Intelligent Electric Vehicles. *SAE Tech. Pap.* **2018**. [CrossRef]
34. Wang, Z.; Han, K.; Tiwari, P. Digital twin simulation of connected and automated vehicles with the unity game engine. In Proceedings of the 2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI), Beijing, China, 15 July–15 August 2021; pp. 180–183. [CrossRef]
35. Elsisi, M.; Tran, M.Q. Development of an IoT Architecture Based on a Deep Neural Network against Cyber Attacks for Automated Guided Vehicles. *Sensors* **2021**, *21*, 8467. [CrossRef]
36. Liu, J.; Dong, Y.; Liu, Y.; Li, P.; Liu, S.; Wang, T. Prediction Study of the Heavy Vehicle Driving State Based on Digital Twin Model. In Proceedings of the 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA 2021), Shenyang, China, 22–24 January 2021; pp. 789–797. [CrossRef]
37. Bottani, E.; Cammardella, A.; Murino, T.; Vespoli, S. From the Cyber-Physical System to the Digital Twin: The process development for behaviour modelling of a Cyber Guided Vehicle in M2M logic. In Proceedings of the XXII Summer School Francesco TurcoIndustrial Systems Engineering, Brescia, Italy, 8–10 September 2021.
38. Guerra, R.H.; Quiza, R.; Villalonga, A.; Arenas, J.; Castano, F. Digital Twin-Based Optimization for Ultraprecision Motion Systems with Backlash and Friction. *IEEE Access* **2019**, *7*, 93462–93472. [CrossRef]
39. Singh, S.; Weeber, M.; Birke, K.P. Implementation of Battery Digital Twin: Approach, Functionalities and Benefits. *Batteries* **2021**, *7*, 78. [CrossRef]
40. Wu, B.; Widanage, W.D.; Yang, S.; Liu, X. Battery digital twins: Perspectives on the fusion of models, data and artificial intelligence for smart battery management systems. *Energy AI* **2020**, *1*, 100016. [CrossRef]
41. Guo, J.; Li, Y.; Pedersen, K.; Stroe, D.I. Lithium-Ion Battery Operation, Degradation, and Aging Mechanism in Electric Vehicles: An Overview. *Energies* **2021**, *14*, 5220. [CrossRef]
42. Li, W.; Rentemeister, M.; Badeda, J.; Jöst, D.; Schulte, D.; Sauer, D.U. Digital twin for battery systems: Cloud battery management system with online state-of-charge and state-of-health estimation. *J. Energy Storage* **2020**, *30*, 101557. [CrossRef]
43. Sancarlos, A.; Cameron, M.; Abel, A.; Cueto, E.; Duval, J.L.; Chinesta, F. From ROM of Electrochemistry to AI-Based Battery Digital and Hybrid Twin. *Arch. Comput. Methods Eng.* **2021**, *28*, 979–1015. [CrossRef]

44. Soleymani, A.; Maltz, W. Real Time Prediction of Li-Ion Battery Pack Temperatures in EV Vehicles. In Proceedings of the ASME 2020 International Technical Conference and Exhibition on Packaging and Integration of Electronic and Photonic Microsystems, Online, 27–29 October 2020. [CrossRef]
45. Wang, W.; Wang, J.; Tian, J.; Lu, J.; Xiong, R. Application of Digital Twin in Smart Battery Management Systems. *Chin. J. Mech. Eng.* **2021**, *341*, 34. [CrossRef]
46. Sulzer, V.; Marquis, S.G.; Timms, R.; Robinson, M.; Chapman, S.J. Python Battery Mathematical Modelling (PyBaMM). *J. Open Res. Softw.* **2021**, *9*, 1–8. [CrossRef]
47. Peng, Y.; Zhang, X.; Song, Y.; Liu, D. A low cost flexible digital twin platform for spacecraft lithium-ion battery pack degradation assessment. In Proceedings of the 2019 IEEE International Instrumentation and measurement technology conference (I2MTC), Auckland, New Zealand, 20–23 May 2019. [CrossRef]
48. Herring, P.; Gopal, C.B.; Aykol, M.; Montoya, J.H.; Anapolsky, A.; Attia, P.M.; Gent, W.; Hummelshøj, J.S.; Hung, L.; Kwon, H.K. BEEP: A Python library for Battery Evaluation and Early Prediction. *SoftwareX* **2020**, *11*, 100506. [CrossRef]
49. Dianov, A. Optimized Field-Weakening Strategy for Control of PM Synchronous Motors. In Proceedings of the 2022 29th International Workshop on Electric Drives: Advances in Power Electronics for Electric Drives (IWED), Moscow, Russia, 26–29 January 2022; pp. 1–6. [CrossRef]
50. Dianov, A.; Anuchin, A. Design of Constraints for Seeking Maximum Torque per Ampere Techniques in an Interior Permanent Magnet Synchronous Motor Control. *Mathematics* **2021**, *9*, 2785. [CrossRef]
51. Venkatesan, S.; Manickavasagam, K.; Tengenkai, N.; Vijayalakshmi, N. Health monitoring and prognosis of electric vehicle motor using intelligent-digital twin. *IET Electr. Power Appl.* **2019**, *13*, 1328–1335. [CrossRef]
52. Rassolkin, A.; Rjabtsikov, V.; Vaimann, T.; Kallaste, A.; Kuts, V.; Partyshev, A. Digital Twin of an Electrical Motor Based on Empirical Performance Model. In Proceedings of the 2020 XI International Conference on Electrical Power Drive Systems (ICEPDS), St. Petersburg, Russia, 4–7 October 2020. [CrossRef]
53. Goraj, R. Digital twin of the rotor-shaft of a lightweight electric motor during aerobatics loads. *Aircr. Eng. Aerosp. Technol.* **2020**, *92*, 1319–1326. [CrossRef]
54. Proksch, D.; Stutz, L.; Krotsch, J.; Hofig, B.; Kley, M. Development of a Digital Twin for an Induction Motor Bearing Voltage Simulation. In Proceedings of the 2021 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan, 16–18 November 2021; pp. 103–107. [CrossRef]
55. Xia, J.; Wang, L.; Zhou, Y. Design and Development of Digital Twin System for Intelligent Motor. In Proceedings of the 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), Dalian, China, 25–27 August 2020; pp. 991–996. [CrossRef]
56. Ruba, M.; Nemes, R.O.; Ciornei, S.M.; Martis, C.; Bouscayrol, A.; Hedesiu, H. Digital twin real-time fpga implementation for light electric vehicle propulsion system using EMR organization. In Proceedings of the 2019 IEEE Vehicle Power and Propulsion Conference (VPPC), Hanoi, Vietnam, 14–17 October 2019. [CrossRef]
57. Abbate, R.; Caterino, M.; Fera, M.; Caputo, F. Maintenance Digital Twin using vibration data. *Procedia Comput. Sci.* **2022**, *200*, 546–555. [CrossRef]
58. Bouzid, S.; Viarouge, P.; Cros, J. Real-Time Digital Twin of a Wound Rotor Induction Machine Based on Finite Element Method. *Energies* **2020**, *13*, 5413. [CrossRef]
59. Ibrahim, M.; Rjabtsikov, V.; Jegorov, S.; Rassolkin, A.; Vaimann, T.; Kallaste, A. Conceptual Modelling of an EV-Permanent Magnet Synchronous Motor Digital Twin. In Proceedings of the 2022 IEEE 20th International Power Electronics and Motion Control Conference (PEMC), Brasov, Romania, 25–28 September 2022; pp. 156–160. [CrossRef]
60. Anwar, M.; Alam, M.K.; Gleason, S.E.; Setting, J. Traction power inverter design for EV and HEV applications at general motors: A review. In Proceedings of the 2019 IEEE Energy Conversion Congress and Exposition (ECCE), Baltimore, MD, USA, 29 September–3 October 2019; pp. 6346–6351. [CrossRef]
61. Mangkalajarn, S.; Ekkaravarodome, C.; Sukanna, S.; Bilsalam, A.; Jirasereeamongkul, K.; Higuchi, K. Comparative Study of Si IGBT and SiC MOSFET in Optimal Operation Class-E Inverter for Domestic Induction Cooker. In Proceedings of the 2019 Research, Invention, and Innovation Congress (RI2C), Bangkok, Thailand, 11–13 December 2019. [CrossRef]
62. Hirao, T.; Onishi, M.; Yasuda, Y.; Namba, A.; Nakatsu, K. EV Traction Inverter Employing Double-Sided Direct-Cooling Technology with SiC Power Device. In Proceedings of the 2018 International Power Electronics Conference (IPEC-Niigata 2018-ECCE Asia), Niigata, Japan, 20–24 May 2018; pp. 2082–2085. [CrossRef]
63. Lu, J.; Hou, R.; Di Maso, P.; Styles, J. A GaN/Si Hybrid T-Type Three-Level Configuration for Electric Vehicle Traction Inverter. In Proceedings of the 2018 IEEE 6th Workshop on Wide Bandgap Power Devices and Applications (WiPDA), Atlanta, GA, USA, 31 October–2 November 2018; pp. 77–81. [CrossRef]
64. Milton, M.; De La, C.O.; Ginn, H.L.; Benigni, A. Controller-Embeddable Probabilistic Real-Time Digital Twins for Power Electronic Converter Diagnostics. *IEEE Trans. Power Electron.* **2020**, *35*, 9852–9866. [CrossRef]
65. Wunderlich, A.; Santi, E. Digital twin models of power electronic converters using dynamic neural networks. In Proceedings of the 2021 IEEE Applied Power Electronics Conference and Exposition (APEC), Phoenix, AZ, USA, 14–17 June 2021; pp. 2369–2376. [CrossRef]

66. Liu, Y.; Chen, G.; Liu, Y.; Mo, L.; Qing, X. Condition Monitoring of Power Electronics Converters Based on Digital Twin. In Proceedings of the 2021 IEEE 3rd International Conference on Circuits and Systems (ICCS), Chengdu, China, 29–31 October 2021; pp. 190–195. [CrossRef]
67. Wu, Q.; Wang, W.; Wang, Q.; Xiao, L.; Hu, B. Digital Twin Approach for Degradation Parameters Identification of a Single-Phase DC-AC Inverter. In Proceedings of the 2022 IEEE Applied Power Electronics Conference and Exposition (APEC), Houston, TX, USA, 20–24 March 2022; pp. 1725–1730. [CrossRef]
68. Shi, H.; Xiao, L.; Wu, Q.; Wang, W. Digital Twin Approach for IGBT Parameters Identification of a Three-Phase DC-AC Inverter. In Proceedings of the 2022 IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific), Haining, China, 28–31 October 2022; pp. 1–4. [CrossRef]
69. Liu, X.; Hofmann, M.; Streit, F.; Maerz, M. Digital Twin for Intelligent and SiC-based Drive Systems. In Proceedings of the 2021 11th International Electric Drives Production Conference (EDPC), Erlangen, Germany, 7–9 December 2021. [CrossRef]
70. Xu, M.; Ng, W.C.; Lim, W.Y.B.; Kang, J.; Xiong, Z.; Niyato, D.; Yang, Q.; Shen, X.S.; Miao, C. A Full Dive into Realizing the Edge-enabled Metaverse: Visions, Enabling Technologies, and Challenges. *arXiv* **2022**, arXiv:2203.05471. [CrossRef]
71. Hyun, J.; Choi, H.; Kim, J. Deriving Improvement Plans through Metaverse Technology and Implications. *Int. J. Intell. Syst. Appl. Eng.* **2022**, *10*, 197–204. Available online: https://ijisae.org/index.php/IJISAE/article/view/2257 (accessed on 18 November 2022).
72. Garg, R. Complex machine validations performed with multiphysics simulation: Intelligent performance engineering provides improvements in simulation, design and connectivity for machine builders. *Plant Eng.* **2021**, *75*, 40–43. Available online: https://go.gale.com/ps/i.do?p=AONE&sw=w&issn=0032082X&v=2.1&it=r&id=GALE%7CA667435375&sid=googleScholar&linkaccess=fulltext (accessed on 18 November 2022).
73. Jung, S.; Ferber, S.; Cramer, I.; Bronner, W.; Wortmann, F. Bosch IoT Suite: Exploiting the Potential of Smart Connected Products. *Connect. Bus.* **2021**, 267–282. [CrossRef]

Appendix 3

Publication III

M. Ibrahim, V. Rjabtšikov, A. Rassõlkin, T. Vaimann and A. Kallaste, “Validation of an EV-Permanent Magnet Synchronous Motor Model Based on Analytical Dynamic Approach,” 2022 International Conference on Electrical Machines (ICEM), Valencia, Spain, 2022, pp. 2384–2390, doi: 10.1109/ICEM51905.2022.9910755.

Validation of an EV-Permanent Magnet Synchronous Motor Model Based on Analytical Dynamic Approach

M. Ibrahim, V. Rjabtšikov, A. Rassõlkin, T. Vaimann, and A. Kallaste

2022 International Conference on Electrical Machines (ICEM) | 978-1-6654-1432-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICEM51905.2022.9910755

***Abstract* -- In the electric automotive industry, manufacturers usually do not declare the electrical parameters of electric vehicle (EV) motors. In advanced control systems, accurate knowledge of motor parameters is essential in order to achieve high dynamic performance. Conventional tests for parameter estimation might be risky not only to the motor windings but also to the measurement devices. In this research study, a dynamic analytical approach is used to estimate the electrical parameters of a three-phase permanent magnet synchronous motor PMSM used in an autonomous electric vehicle (AEV). A detailed d-q mathematical model of the PMSM model was presented. No-load/on load tests were experimentally performed on the motor. The motor simulation dynamic model was built using MATLAB/Simulink. Ant lion optimization (ALO) search algorithm was implemented within the motor simulation model to search proper parameters that achieve equality between experimental and simulated d-q current values. The obtained results showed high agreement between simulation and experimental results.**

Index Terms— Permanent magnet synchronous motor, System Validation, Analytical model.

I. Nomenclature

V_d, V_q	d- and q-axis stator voltage components
ψ_d, ψ_q	d- and q-axis flux linkage components
ψ_{pm}	Permanent magnet flux linkages
i_d, i_q	d- and q-axis stator current components
ω_e	Electrical angular velocity
p	Number of poles pairs
L_d , L_q	d- and q-axis inductances
R_s	Stator winding resistance
E_d, E_q	d- and q-axis induced EMF components
T_e , T_l	Motor electrical torque, mechanical load torque

The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of autonomous electric vehicle".

M. Ibrahim, Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, Tallinn 19086, Estonia (mahmoh@taltech.ee)

V. Rjabtšikov, Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, Tallinn 19086, Estonia (viktor.rjabtsikov@taltech.ee)

A. Rassõlkin, Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, Tallinn 19086, Estonia (anton.rassolkin@taltech.ee)

T. Vaimann, Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, Tallinn 19086, Estonia (toomas.vaimann@taltech.ee)

A. Kallaste, Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, Tallinn 19086, Estonia (ants.kallaste@taltech.ee)

II. Introduction

Permanent magnet synchronous motor (PMSM) drives are used in a large variety of applications due to their dynamic performance, high power density, and higher efficiency, which qualify them for wide use in electric vehicle (EV) propulsion [1][2].

To develop a control system for a motor with good control quality, accurate motor parameters are required to allow the correct setting of the controller [3]. The electrical parameters of the EV motor are not usually available by the vehicle manufacturer.

Conventional method used to measure PMSM parameters is confined to open-circuit voltage and short circuit tests[4]. For high-power machines such as EV motors, those tests might be risky not only for the motor winding but also for the traction drive, not to mention the long time and high-power consumption.

On the other hand, there are various verification and validation methods used for PMSM parameters estimation. These methods are based, for example, on steady-state characteristics measurement [5], current and voltage at constant velocity measurement [6], or determination of motor time characteristics [7]. Some motor inverter manufacturers provide the auto-tuning algorithms of the control systems based on static and dynamic characteristics. An overview of PMSM parameters estimation methods was addressed in [8]. Extended Kalman filter combined with gradient correction methods were used for real time PMSM linear electrical parameters estimation[9]. An offline parameters estimation technique of three phase surface mounted PMSM based on exponential current transient was proposed [10]. Validation of the finite element model of a three-phase PMSM based on the numerical solution of Poisson's equation governing the magnetic field problem was presented in [11]. The optimization numerical Box's method was applied in offline parametric identification of the PMSM mathematical model, which was determined according to the minimization of the mean-square error of the stator current and the angular velocity in [3]. Electrical and mechanical parameters of PMSM were identified based on a model reference adaptive system, and simulated annealing particle swarm optimization was illustrated in [12]. An immune clonal differential evolution algorithm (ICDEA) was used to identify the electrical parameters of PMSM proposed in [13]. The proposed method for PMSM parameters estimation is a combination between system identification theory and the optimization search method [8]. It's based on identifying the unknown parameters of the motor mathematical model using known values (such as stator voltages and currents). The motor mathematical model is translated into simulation model. An optimization search algorithm is implemented in the motor simulation model to searched for proper parameter values (unknowns) that achieve equality between experimental and simulation motor known values.

This paper is organized as follows: Section II presents a brief introduction supported by an overview of previous studies. A detailed derivation of the PMSM mathematical d-q model and the equivalent circuit is proposed in Section III. Section IV describes the motor experimental model (test bench) and the procedure of tests done on the motor. Section V proposes the motor dynamic simulation model built using MATLAB/Simulink. An optimization search algorithm is implemented in the motor simulation model to estimate its parameters. The parameter estimation technique compares the values of experimentally measured d-q stator currents with the simulation result. Section VI discusses the main results of the simulation. Conclusions of the research and future plans are addressed in Section VII.

III. EV-Based PMSM Dynamic Mathematical Model

The mathematical model can be constructed on a stationary or rotating reference frame. The proposed method gives better results in rotating reference frame because the model coefficients can be assumed to be almost constant.

Fig. 1 illustrates the PMSM equivalent circuit [14], from which the following mathematical model is deduced.

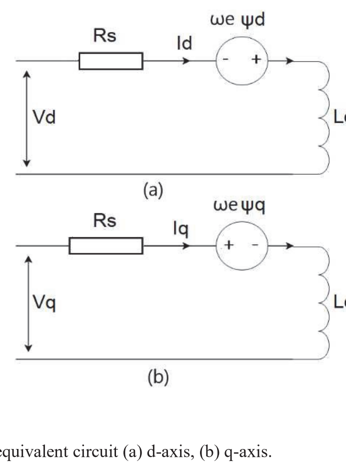


Fig. 1 PMSM equivalent circuit (a) d-axis, (b) q-axis.

The stator voltage equations can be written in the synchronous *d–q* reference frame as follows [15]:

$$V_d = R_s i_d + \frac{d}{dt}\psi_d - \omega_e \psi_q, \tag{1}$$

$$V_q = R_s i_q + \frac{d}{dt}\psi_q + \omega_e \psi_d, \tag{2}$$

whereas the flux linkage equation can be given as

$$\psi_d = \psi_{pm} + L_d i_d, \tag{3}$$

$$\psi_q = L_q i_q. \tag{4}$$

So, the stator voltage equations can be rewritten as follows:

$$V_d = R_s i_d + L_d \frac{d}{dt} i_d + E_d, \tag{5}$$

$$V_{qs} = R_s i_q + L_q \frac{d}{dt} i_q + E_q, \tag{6}$$

where

$$E_d = -\omega_e L_q i_q, \tag{7}$$

$$E_q = \omega_e\left(\psi_{pm} + L_d i_d\right). \tag{8}$$

The electromagnetic torque equation can be defined as follows:

$$T_e = \frac{3}{2}\, p\, i_q\left(\psi_{pm} + \ (L_d - L_q) i_d\right) \tag{9}$$

The mechanical equation of the motor is driven from the general machine swing equation as follows:

$$J\frac{d\omega_e}{dt} * \frac{2}{p} = T_e - T_l \tag{10}$$

Where: J is the motor inertia.

IV. Experimental work

A. Test bench

ISEAUTO autonomous electric vehicle (AEV) was built on a Mitsubishi i-MiEV trolley based on a Y4F1 PMSM [16]. The PMSM of the ISEAUTO was set as a case study. The motor was driven by an EV Inverter (ABB HES880) with the Direct Torque Control (DTC) algorithm. PMSM was equipped with a resolver positioned on its shaft. The EV inverter was fed by a battery emulator system (Cinergia) to behave like real batteries. The inverter was controlled by a PC interface unit. A frequency converter (ABB ACS800) DTC – 32 Kw induction motor (IM) setup was used as a load belt drive. A modular power analyzer and data acquisition system (DEWETRON) was used to measure PMSM's stator input voltages, currents, input power, electrical angle, and, rotor speed. It was also used to measure and acquire the output power of loading IM so as to get the applied load torque profile. All the experiments were performed in a real-time setup, including motors and frequency converters. All the measurements were done at a sampling frequency of 10 kHz. Fig. 2 shows the full test bench of the PMSM under study.

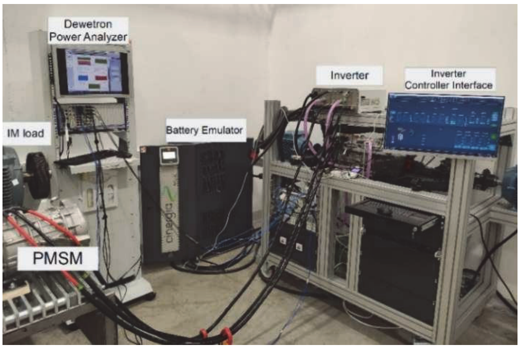


Fig. 2 ISEAUTO-PMSM test bench.

As mentioned previously, the electrical parameters of the motor are unknown; by contrast, the motor ratings known by the manufacturers are usually found on the nameplate. Table 1 describes the PMSM nameplate.

TABLE I. PMSM NAMEPLATE.

Parameter	Description	Value	Unit
p	Number of pole pairs	4	-
N_r	Rated speed	3000	rpm
P_r	Rated output power	35	kW
P_{Max}	Maximum output power	49	kW
T_{Max}	Maximum torque	180	N·m
J	motor inertia	0.00047	kg·m^2

B. *The procedure of experimental tests*

The proposed approach of parameter determination of the PMSM model is based on equality between experimental and simulation d-q currents at the same speed /loading condition. No-load / on load tests were performed under three operating speeds within the motor rated speed range of (300,600,900 rpm). The loading tests were done under four different loading conditions for each speed of no load, light load, medium load and heavy load according to table II. The loading motor was driven under DTC algorithm which enables torque regulation as a percent of the rated machine torque (T_{rated} = 128.4 N.m).

TABLE II. PMSM NAMEPLATE.

Load Torque condition	load percent
No load	1%
Light load	15%
Medium load	30%
Heavy load	60%

The measured data from experimental tests contains some harmonics at an acceptable level, as expected due to the noises resulted from voltage and current sensors and also the inverter switching mechanism. To eliminate those harmonics, a virtual first-order filter (DEWETRONE function) was used. The First Order Filter is useful for purposes such as noise reduction. The filter time constant (τ) was specified in the filter's properties in seconds of 10*e^{-3}s.

Fig. 3 shows an example of the actual and, filtered voltage waveform.

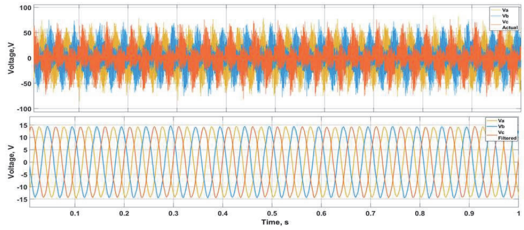


Fig. 3 Voltage waveform actual and, filtered.

All the measured data of each test was stored in the form of a MATLAB data file to facilitate its use in the simulation model discussed in detail in the next section.

V. SIMULATION MODEL

A. *Model Description*

The simulation d-q model is built as a replica of the experimental model (test bench) with extra features, as shown in Fig. 4. The model is able to handle the data measured and collected from the experimental tests. The unknown parameters of the motor are the stator d-q inductances (L_d, L_q) and the permanent magnet linkage flux ψ_{pm}. The stator resistance (R_s) is measured experimentally by a DC test.

Input voltage block is responsible for receiving and filtering stator voltage signals and feeding them to the motor stator. The stator current block receives and filters the experimentally measured stator currents; then the block converts them into the d-q rotating reference frame to make it easier to handle them by the optimization search algorithm. The comparator block is responsible for comparing currents from experimental and simulation models based on the implemented search algorithm in the motor model. Resolver decoder block implements the measured data of resolver windings, converting it to an accurate rotor position. PMSM block implements the mathematical d-q model as shown in Fig. 5.

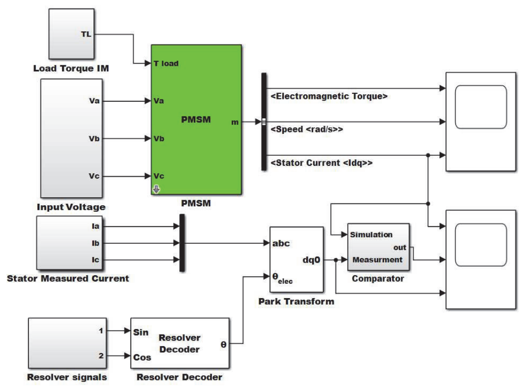


Fig. 4. Simulink model of PMSM testbench.

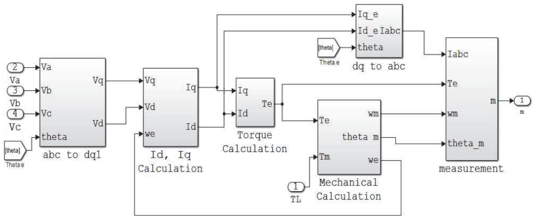
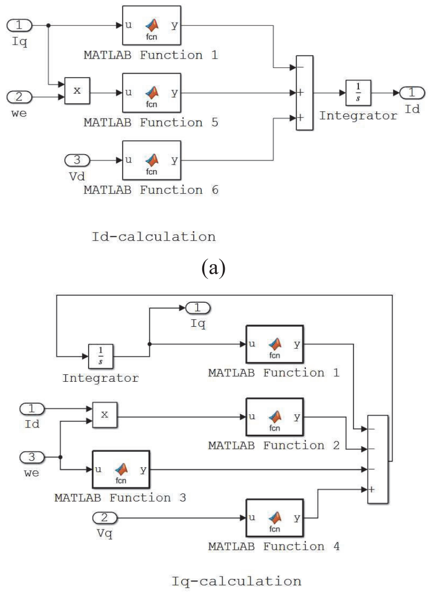


Fig. 5 PMSM expanded block.

Each sub-block inside the PMSM block represents an equation from the mathematical model addressed in Section II. An optimization search algorithm is implemented in the I_d, I_q calculation block where the effect of unknown parameters is high. The algorithm is done as a MATLAB script connected to the Simulink model with a function block, as shown in Fig. 6.



(a)

(b)

Fig. 6 Expanded block of i_d (a) and i_q (b) calculation.

B. ALO Search Algorithm

Ant Lion Optimization (ALO) algorithm was developed by Seyedali Mirjalili as a meta-heuristic optimization technique to simulate the hunting mechanism of ant lions' predators with their favorite ants in nature [17]. The algorithm starts by generating an ant population, the fitness function, and updating ant and ant lion positions later. It ends by testing the stopping criterion. The general steps describing the ALO technique are summarized as follows.

1- Initialization: Random population of ants is generated that moves around ant lions in the search space.

2- Fitness: The fitness of each ant position is evaluated in the space concerning ant lions.

3- Position Update: Update the position of each and use random walk concerning the ant lions based on roulette wheel until the best ant lion is obtained and store it as an elite.

4- Evaluation: Evaluate and update the best new ant lion position to their objective values.

5- Test: If the criterion is achieved, stop, and find the current best ant lion position.

6- Loop: If the number of iterations equals the maximum stop, go to step (2) with the best ant lion position obtained in step (5).

A deductive search program was applied to select L_d, L_q and ψ_{pm} that achieves equality between experimentally measured and simulated stator d-q currents.

The objective function F with equalities constraints is given as follows:

(a) objective function:

$$F = [\min\{L_d, L_q, \psi_{pm}\}]; \tag{11}$$

(b) optimization equalities:

$$\begin{cases} i_{d(experimental)} = i_{d(simulation)} \\ i_{q(experimental)} = i_{q(simulation)} \end{cases}; \tag{12}$$

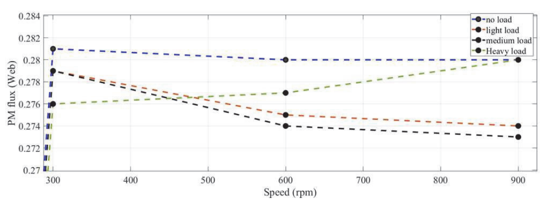
(c) constraints:

$$\begin{gathered} 0 < L_q < 0.1 \\ L_d < L_q \\ 0 < \psi_{pm} < 1 \end{gathered} \tag{13}$$

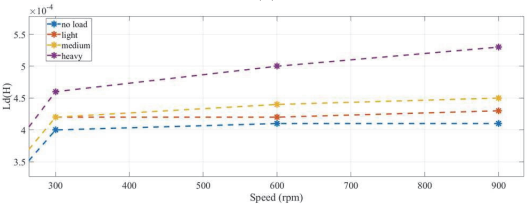
VI. Simulation Results and Discussion

Simulation tests were carried out according to the previously performed experimental tests. The simulation model inputs are stator voltages, load torque (T_L), and rotor position (θ_m) taken from experimental tests measurements. At the same time, the outputs are rotor speed, stator currents, and electromagnetic torque (T_e). A park transformation block is used to transform the measured experimental three-phase stator current into a d-q reference frame to be set as a comparison reference for the ALO search algorithm. Simulation was carried out in two stages, the searching and comparison stage, followed by the simulation of the selected results stage. The results obtained by ALO showed some slight differences in the parameter values according to different speed and loading conditions, as shown in Fig. 7.

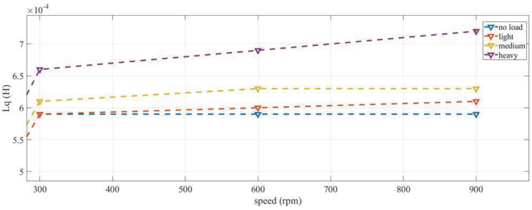
Fig. 7(a) shows that the values of PM linkage flux range between 0.274 and 0.281 Wb. At low speed, the value of PM linkage flux is higher than at higher speed, which could be explained by the flux weakening algorithm of the EV- inverter with high speeds.



(a)



(b)



(c)

Fig. 7 ALO results of (a) PM linkage flux, (b) L_d, (c) L_q.

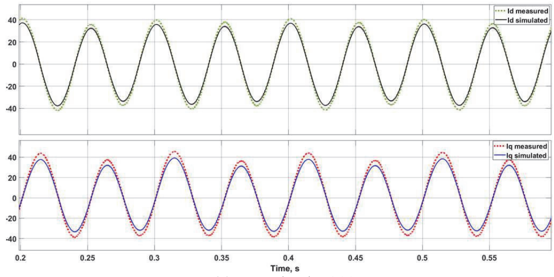
As can be seen from Fig. 7(b), the variation in the values of L_d ranges between 0.40 and 0.53 mH, while in Fig. 7(c), L_q ranges between 0.59 and 0.66 mH. Thus, the percentage difference between the smallest and the largest value in all cases did not exceed 10%, which is acceptable. Therefore, with these superficial differences between the values obtained through ALO, the search circle has become much smaller; therefore, resorting to the average value is logical.

Table III provides detailed parameters data for each test, resulting from ALO and the average value of each obtained parameter.

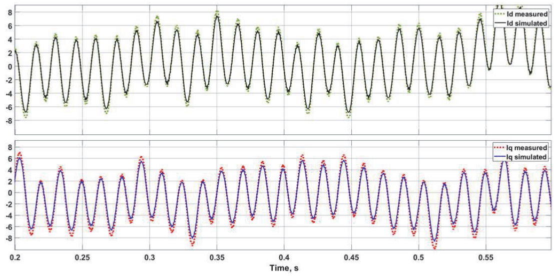
TABLE III. TEST RESULTS FROM ALO ALGORITHM

Condition	Speed(rpm)	L_d(mH)	L_q(mH)	ψ_{pm}(Wb)
No load	300	0.40	0.59	0.281
	600	0.41	0.59	0.280
	900	0.41	0.59	0.280
Light load	300	0.42	0.59	0.279
	600	0.43	0.60	0.276
	900	0.43	0.61	0.274
Medium load	300	0.42	0.61	0.276
	600	0.43	0.63	0.277
	900	0.45	0.66	0.280
Heavy load	300	0.46	0.66	0.276
	600	0.50	0.69	0.279
	900	0.53	0.72	0.281
Average		0.44	0.57	0.278

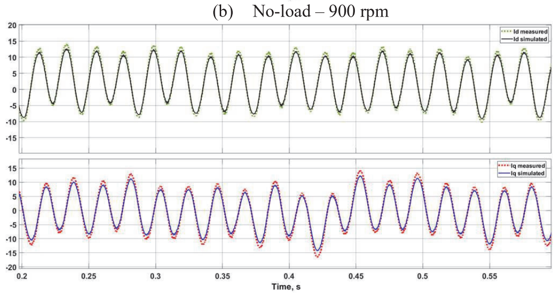
After obtaining motor parameters, in the verification step, the resulting simulation stator d-q currents are compared with those experimentally measured. The simulation is carried out for all the loading and speed cases previously experimentally measured. Most of the cases show high agreement between the experimental and the simulation values of stator d-q currents. By contrast, some other cases showed some differences, which can be explained by the behavior of the EV inverter algorithm. Fig. 8 shows the agreement between the measured and the simulation d-q stator current under different loading and speeds.



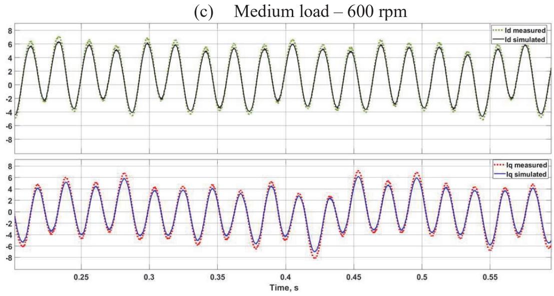
(a) No-load – 300 rpm



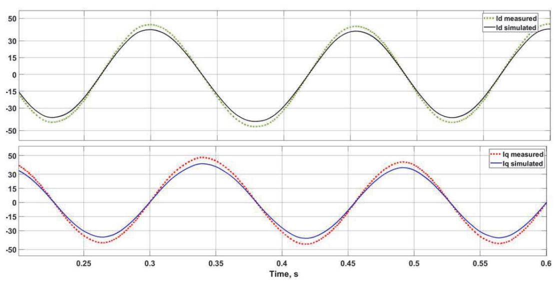
(b) No-load – 900 rpm



(c) Medium load – 600 rpm



(d) Light load – 600 rpm



(e) Heavy load - 300 rpm

Fig. 8. Comparison between simulation and experimental results.

VII. Conclusion

This study proposed an effective methodology for attaining electrical parameters of an EV- PMSM based on an analytical approuch. Experimental no-load/ loading tests with specific conditions were performed on the motor. An analytical simulation d-q motor model was built using Matlab/ Simulink. ALO search algoritm was implemented in the motor simulation model.

The proposed methodology relied on searching for the appropriate parameter values that achieve the highest degree of equality between the measured and simulated d-q currents.

The obtained motor parameters were used in the motor simulation model. Simulation results showed high agreement with experimental results, which proves the effectiveness of the proposed methodology.

Unlike the conventional methods for PMSM parameters estimation, the proposed methodology is

1- Simple: It required normal no load and loading tests.
2- Fast response time: The used analytical d-q model provides faster response time than other modelling methods.
3- High accuracy: The convergence of the obtained parameter values with different testing conditions proves the accuracy of the proposed method.

For future work, the proposed PMSM model can be extended to include the effect core losses resistance in the motor mathmatical model so as to estimate its value.

VIII. Acknowledgment

The authors gratefully acknowledge the contributions of M.U. Naseer for his work on the original version of this document.

IX. References

[1] R. Morales-Caporal, M. E. Leal-Lopez, J. De Jesus Rangel-Magdaleno, O. Sandre-Hernandez, and I. Cruz-Vega, "Direct torque control of a PMSM-drive for electric vehicle applications," *2018 28th Int. Conf. Electron. Commun. Comput. CONIELECOMP 2018*, vol. 2018-January, pp. 232–237, Mar. 2018, doi: 10.1109/CONIELECOMP.2018.8327204.

[2] A. Dianov, "Highly Efficient Sensorless Multicontrol Mode Compressor IPMSM Drive With Seamless Transitions," *IEEE Trans. Power Electron.*, vol. 37, no. 8, pp. 9129–9137, Aug. 2022, doi: 10.1109/TPEL.2022.3157476.

[3] L. Zawarczynski and T. Stefanski, "Parametric identification of PMSM mathematical model," *2019 24th Int. Conf. Methods Model. Autom. Robot. MMAR 2019*, pp. 478–483, Aug. 2019, doi: 10.1109/MMAR.2019.8864702.

[4] M. Kondo, "Parameter measurements for permanent magnet synchronous machines," *IEEJ Trans. Electr. Electron. Eng.*, vol. 2, no. 2, pp. 109–117, Mar. 2007, doi: 10.1002/TEE.20117.

[5] A. Cavagnino, M. Lazzari, F. Profumo, and A. Tenconi, "Axial flux interior PM synchronous motor: parameters identification and steady-state performance measurements," *Conf. Rec. - IAS Annu. Meet. (IEEE Ind. Appl. Soc.*, vol. 4, pp. 2552–2559, 1999, doi: 10.1109/IAS.1999.799199.

[6] F. Mink, N. Kubasiak, B. Ritter, and A. Binder, "Parametric model and identification of PMSM considering the influence of magnetic saturation," *Proc. Int. Conf. Optim. Electr. Electron. Equipment, OPTIM*, pp. 444–452, 2012, doi: 10.1109/OPTIM.2012.6231768.

[7] M. Pulvirenti, G. Scarcella, G. Scelba, A. Testa, and M. M. Harbaugh, "On-line stator resistance and permanent magnet flux linkage identification on open-end winding PMSM Drives," *IEEE Trans. Ind. Appl.*, vol. 55, no. 1, pp. 504–515, Jan. 2019, doi: 10.1109/TIA.2018.2869877.

[8] R. P., M. K. ALICE, K. M. SURYA, and S. A., "Parameter Estimation Of Permanent Magnet Synchronous Motor-A Review," *i-manager's J. Electr. Eng.*, vol. 9, no. 2, p. 49, 2015, doi: 10.26634/JEE.9.2.3719.

[9] Y. Zhang, Y. Bi, and S. Wang, "Parameter Identification of Permanent Magnet Synchronous Motor Based on Extended Kalman Filter and Gradient Correction," *2020 IEEE Int. Conf. Mechatronics Autom. ICMA 2020*, pp. 718–722, Oct. 2020, doi: 10.1109/ICMA49215.2020.9233766.

[10] A. Dianov, "An algorithm for estimation of stator resistance and inductance of low-cost SMPMSM drives," *IEEE J. Emerg. Sel. Top. Power Electron.*, 2022, doi: 10.1109/JESTPE.2022.3160154.

[11] A. Saadoun and Y. Amirat, "Validation of a PMSM model based on static and transient FEM," *Int. Conf. Power Eng. Energy Electr. Drives*, pp. 1130–1134, 2013, doi: 10.1109/POWERENG.2013.6635769.

[12] G. Su, P. Wang, Y. Guo, G. Cheng, S. Wang, and D. Zhao, "Multiparameter Identification of Permanent Magnet Synchronous Motor Based on Model Reference Adaptive System—Simulated Annealing Particle Swarm Optimization Algorithm," *Electron. 2022, Vol. 11, Page 159*, vol. 11, no. 1, p. 159, Jan. 2022, doi: 10.3390/ELECTRONICS11010159.

[13] G. Lin, J. Zhang, and Z. Liu, "Parameter identification of PMSM using immune clonal selection differential evolution algorithm," *Math. Probl. Eng.*, vol. 2014, 2014, doi: 10.1155/2014/160685.

[14] A. N. Reshetnikov and E. V. Reshetnikova, "PMSM analysis in dq axis at starter mode as a part of ISG," *Int. Conf. Young Spec. Micro/Nanotechnologies Electron Devices, EDM*, vol. 2018-July, pp. 620–622, Aug. 2018, doi: 10.1109/EDM.2018.8435057.

[15] N. A. Elsonbaty, M. A. Enany, and M. I. Hassanin, "An Efficient Vector Control Policy for EV-Hybrid Excited Permanent-Magnet Synchronous Motor," *World Electr. Veh. J. 2020, Vol. 11, Page 42*, vol. 11, no. 2, p. 42, May 2020, doi: 10.3390/WEVJ11020042.

[16] A. Rassõlkin, R. Sell, and M. Leier, "Development Case Study of the First Estonian Self-Driving Car, ISEAUTO," *Electr. Control Commun. Eng.*, vol. 14, no. 1, pp. 81–88, 2018, doi: 10.2478/ecce-2018-0009.

[17] S. Mirjalili, "The Ant Lion Optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, May 2015, doi: 10.1016/J.ADVENGSOFT.2015.01.010.

X. Biographies

Mahmoud Ibrahim was born in Zagazig, Egypt, in 1993. He received his B.Sc. and MSc. degrees in Electric Power and Machines from Zagazig University, Egypt, in 2016, and 2021 respectively, and the ongoing Ph.D. degree from the Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology (TalTech), Estonia. His research interests include digital twins, modeling and simulations, optimization, electric machines control systems, and electric vehicles propulsion drives.

Viktor Rjabtšikov received his BSc and MSc degrees in electrical engineering in 2018 and 2020, respectively. He is currently a PhD Student at Department of Electrical Power Engineering and Mechatronics in Tallinn University of Technology. In addition, he is a Branch Secretary of The IEEE Student Branch at Tallinn University of Technology. Viktor has participated in various internships at different Estonian companies. Primary research interest relates to electrical drives and electrical machine control theory. His research interest is electrical vehicles propulsion system and Digital Twins implementation to EV systems.

Anton Rassõlkin was born in Tallinn, Estonia, in 1985. He received the BSc, MSc, and PhD degrees in electric drives and power electronics from Tallinn University of Technology (Estonia) in 2008, 2010, and 2014, respectively. In 2010 he received a Dipl.-Ing. degree in automatic from the University of Applied Science Giessen-Friedberg (Germany).
He has been working in several companies as an electrical engineer and universities as a lecturer. Presently, he holds the position of professor in Mechatronics at the Department of Electrical Power Engineering and Mechatronics, School of Engineering, Tallinn University of Technology (TalTech). Moreover, he serves as a visiting professor at the Faculty of Electrical Engineering Department of Power Electronics, Electrical Drives and Robotics at Silesian University of Technology (Gliwice, Poland).
The main research interests are mechatronics and electrical drives, particularly for electric transportation, as well as autonomous vehicles.
He is a Senior Member of IEEE and the Estonian Society of Moritz Hermann Jacobi.

Toomas Vaimann was born in Pärnu, Estonia, in 1984 and received his BSc, MSc and PhD degrees in electrical engineering from Tallinn University of Technology, Estonia, in 2007, 2009 and 2014 respectively. He is currently a senior researcher in Tallinn University of Technology, Department of Electrical Engineering. Internationally he has been a postdoctoral researcher at the Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland.
He has been working in several companies as an electrical engineer. He is the member of IEEE (S'11-M'14-SM'20), Estonian Society of Moritz Hermann Jacobi and Estonian Society for Electrical Power Engineering. His main research interest is the diagnostics of electrical machines.

Ants Kallaste was born in Pärnu, Estonia in 1980 and received his BSc, MSc, and PhD degrees in electrical engineering from Tallinn University of Technology, Estonia, in 2004, 2006 and 2013 respectively. He is currently a Professor of Electrical Machines in Tallinn University of Technology, Department of Electrical Power Engineering and Mechatronics. In addition, he is holding the position of Head of Chair of Electrical Machines Research Group. He is the senior researcher of IEEE and member of Estonian Society of Moritz Hermann Jacobi.
He has spent several years in private engineering companies and visited numerous other research institutions. Amongst other research activities, Ants Kallaste is involved with expertise and consultations of private companies in the field of electrical machines, drives, and their diagnostics.

Appendix 4

Publication IV

M. Ibrahim, V. Rjabtšikov, S. Jegorov, A. Rassõlkin, T. Vaimann and A. Kallaste, “Conceptual Modelling of an EV-Permanent Magnet Synchronous Motor Digital Twin,” 2022 IEEE 20th International Power Electronics and Motion Control Conference (PEMC), Brasov, Romania, 2022, pp. 156–160, doi: 10.1109/PEMC51159.2022.9962943.

Conceptual Modelling of an EV-Permanent Magnet Synchronous Motor Digital Twin

Mahmoud Ibrahim
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
mahmoh@taltech.ee

Viktor Rjabtšikov
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
viktor.rjabtsikov@taltech.ee

Sergei Jegorov
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
sejego@taltech.ee

Anton Rassõlkin
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
anton.rassolkin@taltech.ee

Toomas Vaimann
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
toomas.vaimann@taltech.ee

Ants Kallaste
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
ants.kallaste@taltech.ee

2022 IEEE 20th International Power Electronics and Motion Control Conference (PEMC) | 978-1-6654-9681-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/PEMC51159.2022.9962943

***Abstract*— Digital twin (DT) technology has contributed to the development process of many applications, including electric vehicles (EVs). The DT concept is to create a digital representation of a real physical asset and support its performance by utilizing simulation and optimization tools fed with real-time data. DT technology can be used to solve general problems related to EV motors, such as estimation of the driving torque and the internal rotor temperature. This paper provides the concepts for implementing a DT of an EV permanent magnet synchronous motor (PMSM) based on its analytical performance model. DT architecture comprises two main components: virtual model and real-time data exchange set. The motor physical model (test bench) was provided in detail. An analytical performance q-d mathematical model supported by the motor equivalent circuit was explained. The motor virtual model was built based on the proposed analytical model using MATLAB/Simulink. Robot operating system V.2 (ROS2) node, implemented on a microcontroller, was used for real-time data exchange between the physical and the virtual motor models. The main target is to monitor the physical motor performance and estimate its torque through its digital twin. The obtained results from the DT showed the effectiveness of the proposed method.**

Keywords—Digital Twin, Torque Measurement, Permanent magnet synchronous motor.

Nomenclature

u_{sd}, u_{sq}	d- and q-axis stator voltage components
;|$_{sd}$, ;|$_{sq}$	d- and q-axis flux linkage components
;|$_{pm}$	Permanent magnet flux linkages
i_{sd}, i_{sq}	d- and q-axis stator current components
w_e	Electrical angular velocity
p	Number of poles pairs
l_{sd} , l_{sq}	d- and q-axis inductances
r_s	Stator winding resistance
E_{sq}	d- and q-axis induced EMF components
T_e , T_m	Motor electrical torque, mechanical load torque
J	Rotor inertia

I. Introduction

Digital twin (DT), as a definition, is to create and maintain a digital representation of a real physical object, asset, process, or service. It's, in essence, a computer program that uses real-world data to create simulations that can predict how a product or process will perform[1].

While simulations and DTs both use digital models to replicate products and processes, the difference between them is that DT creates a virtual environment able to study several simulations, backed up with real-time data and a two-way flow of information between the twin and the sensors that collect this data. This increases the accuracy of predictive analytical models, offering a greater understanding of the management and monitoring of products, policies, and procedures. DT technology has conquered many applications and sectors such as aerospace, smart cities, electric vehicles (EVs), and industrial robotics[2]. DT for EV includes many functions such as health monitoring, diagnostics, prognostics, optimization, scenario, and risk assessment [3]. It can be created at the system level, subsystem level, component level, and many other assets.

The electric motor is considered the core element of the propulsion system of any EV. It must meet the EV requirements of high performance, high torque/power density, and mainly high operational efficiency [4]. To achieve that, two main conditions must be fulfilled: a proper motor design and an efficient control algorithm. Most EV motors depend on torque-based control algorithms [5]. Torque estimation is critical for EV manufacturers as it is a vital variable in powertrain management and energy-saving strategies. It is preferable to avoid using torque transducers in EV motors for reasons concerning size, cost, and mechanical positioning challenges.

There are many research efforts in EV motors torque estimation techniques as the following pieces of literature show. A non-intrusive approach using nonlinear Kalman filtering methodology to obtain torsional load torque information on mechatronic powertrains[6]. An artificial neural network statistical based method was investigated for PMSM rotor torque estimation based on the motor electrical signals[7]. An extended Kalman filter combined with a second-order model observer was developed to observe the loading torque of an EV switched reluctance motor (SRM) [8].

In a different context, condition monitoring is one of the main usages of the DT. The following pieces of literature show the impact of this DT function. A cross-platform engine Unity 3D was used as a virtual environment of a DT for monitoring an induction motor based on an empirical performance model [1]. Inter-turn short circuit fault detection was implemented into a DT of an AC 3-phase induction motor (IM) in [9]. DT of an EV motor to optimize the motor performance concerning estimating driving torque and cooling control based on a micro lab box was proposed in [10]. Monitoring and conditions analyzing DT of an IM based on a simulation finite element method (FEM) model were addressed in [6]. DT model for fault detection of a 50 MW PMSM based on a numerical analysis model was discussed in [11]. Health monitoring and lifetime prediction of an EV PMSM were done by implementing an intelligent DT model based on MATLAB/Simulink and a mixed fuzzy logic in [12].

This paper proposes the concepts of an EV – PMSM DT functioned for electromagnetic torque estimation. It is organized as follows; An overview of DT technology and its areas of applications for EV propulsion motors is discussed in section I. Section II illustrates the mathematical dynamic model derivation and equivalent circuits of PMSM. Section III presents the main architecture of the motor DT, which is divided into three subsections. The reduced physical model (test bench) of EV PMSM is presented in section III.A. The data exchange set (communication) model between physical and virtual models is proposed in III.B. Section III.C provides the virtual (Simulation) model of the motor. The concept of DT is achieved by liking the virtual and physical model based on ROS2 framework. Obtained preliminary results are discussed in section IV. Conclusion and future works are addressed in section V.

II. EV-PMSM Analytical performance model

In EV applications, the electric motor can be modeled and analyzed in several ways. Depending on the selected analysis technique, the simulation time and results accuracy could vary a lot. There are three main techniques to model an electric motor, and thereafter study its different variables. These methods are: Finite Element Analysis (FEA), *d-q* Equivalent Circuit (*d-q* EC), and Magnetic Equivalent Circuit (MEC). For real-time torque measurement the *d-q* EC is fastest method in terms of response time[13].

Although in experimental motor operation with variable speed drives, several harmonics are produced and observed, yet the first-harmonic *d-q* circuit model is an essential tool for motor performance analysis. It is mainly used to calculate the motor voltage and electromagnetic torque[14].

The stator flux linkage vector of PMSM can be drawn in the rotor reference frame (*d-q*) and, stator reference frame (*α-β*), as shown in Fig.1.

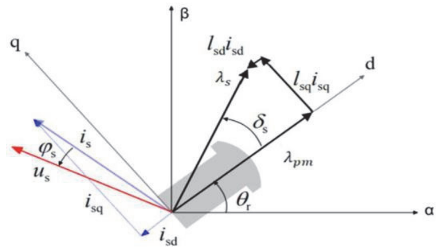


Fig. 1. Vector diagram of a PMSM. Stator reference frame (α-β) and rotor reference frame (d-q).

When the rotor reference frame is considered, the equivalent *d* and *q* axis stator windings are transformed into the reference frames revolving at rotor speed. The consequence is that there is zero speed differential between the rotor and stator magnetic fields, and the stator *d* and *q* axis windings have a fixed phase relationship with the rotor magnet axis, which is the *d* axis in the modeling [15].

Fig. 2 shows the *d-q* equivalent circuit of PMSM from which the following equations are deduced.

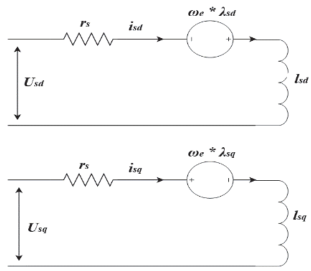


Fig. 2. PMSM d-q equivalent circuit.

The stator voltage equations can be written in synchronous *d–q* reference frame as follows:

$$u_{sd} = r_s i_{sd} + \frac{d}{dt}\lambda_d - w_e\lambda_{sq}, \tag{1}$$

$$u_{sq} = r_s i_{sq} + \frac{d}{dt}\lambda_q + w_e\lambda_{sd}, \tag{2}$$

where the flux linkage equation can be given below.

$$\lambda_{sd} = \lambda_{pm} + l_{sd}i_{sd}, \tag{3}$$

$$\lambda_{sq} = l_{sq}i_{sq}. \tag{4}$$

So, the stator voltage equations can be rewritten as follows:

$$u_{sd} = r_s i_{sd} + l_{sd}\frac{d}{dt}i_{sd} + E_{sd}, \tag{5}$$

$$u_{sq} = r_s i_q + l_q\frac{d}{dt}i_q + E_{sq}, \tag{6}$$

where

$$E_{sd} = -w_e l_{sq} i_{sq}, \tag{7}$$

$$E_{sq} = w_e(\lambda_{pm} + l_{sd}i_{sd}), \tag{8}$$

taking Laplace transformation then $d/dt = s$, then the stator voltage matrix can be expressed as follows:

The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of an autonomous electric vehicle".

$$\begin{bmatrix} u_{sd} \\ u_{sq} \end{bmatrix} = \begin{bmatrix} r_s + sl_{sd} & -w_e l_{sq} \\ w_e l_{sd} & r_s + sl_{sq} \end{bmatrix} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} + \begin{bmatrix} 0 \\ w_e \lambda_{pm} \end{bmatrix} \quad (9)$$

The electromagnetic torque equation can be defined as follows

$$T_e = \frac{3}{2} p\, i_{sq} (\lambda_{pm} + (l_{sd} - l_{sq}) i_{sd}) \quad (10)$$

The mechanical equation of the motor is driven from the general machine swing equation as follows:

$$\frac{(\ast s \quad)\ast 2\ast}{p} = T_e - T_m \quad (11)$$

Where B is the friction coefficient.

III. Digital Twin of EV-PMSM

As previously discussed, that DT is a replica of an existing physical model. It consists of three main parts as follows: physical model, Data exchange set (communication model), and virtual simulation model.

A. *Physical Model (test bench)*

The motor physical model was taken from ISEAUTO to be a real representation of an EV motor. ISEAUTO was built on a Mitsubishi i-MiEV trolley based on a Y4F1 PMSM [16]. Table. 1 present ISEAUTO motor parameters.

TABLE I. Understudy PMSM parameters

Parameter	Description	Value	Unit
p	Number of pole pairs	4	-
rs	Stator resistance	0.07	Ω
lsq	q-axis inductance	4.4·10-4	H
lsd	d-axis inductance	5.7·10-4	H
λ_{pm}	Permanent magnet linkage flux	0.279	Web
Nr	Rated speed	3000	rpm
Pr	Rated output power	35	kW

The motor test bench is set as the following. The motor is driven by an EV Inverter (ABB HES880) with direct torque control (DTC) algorithm. It's also equipped with a resolver for speed and position measurement fixed on its shaft. The EV inverter was fed by a battery emulator system (Cinergia B2C+30). The inverter was driven by a visual interface unit.

Voltage and current sensors were placed on the motor input terminals to collect stator voltage and current input signals. Voltage sensors were also connected to the resolver main and auxiliary windings to collect motor speed and position data. Fig.3 shows the experimental motor test bench.

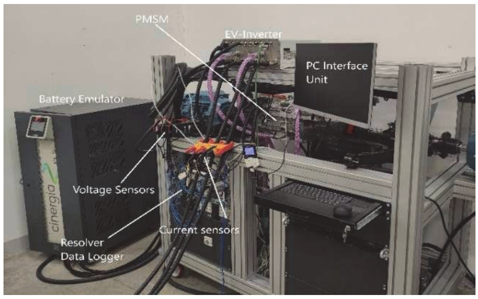


Fig. 3. PMSM test bench.

B. *Data Exchange set (Communication model).*

The outputs of the test bench sensors were connected to ROS2 Foxy node implemented on a (Teensy 4.0) board. Teensy 4.0 is a microprocessor development board manufactured by PJRC. It features ARM Cortex-M7 processor, float point math units, 1984 KB of Flash and 1024 KB RAM memory and features a total of 40 GPIO pins. Real time data are received via the subscriber of MATLAB ROS2 Toolbox. The subscriber is a node that subscribes to a topic and processes the received data. The received data are ROS2 messages from the Digital Twin middleware, and each message is a structure consisting of three-phase currents, three-phase voltages, and resolver signals. Upon reception, the stator phase voltage and resolver signals are extracted by the subscriber and are sent to the simulation model for further analysis and processing. Simulated real time data of motor torque is sent to an interface module (Workstation) to be analyzed. As a future step, the analyzed data is transferred into ta commends to control the real motor model. Fig. 4 shows the operational architect of DT.

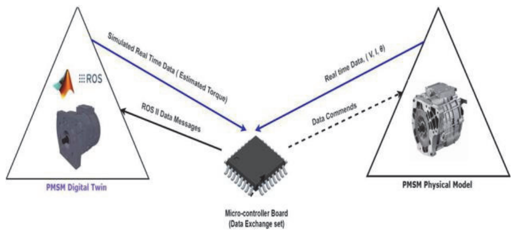


Fig. 4. The operational architecture of PMSM Digital Twin.

C. *Simulation Model*

The motor simulation model was built using MATLAB/Simulink on the basis of PMSM analytical d-q model from section II. PMSM block implements the derived equations of the proposed mathematical model. Two ROS2 subscribers were implemented in the simulation model to enable data exchange with the physical model.

Subscriber 1 collects real-time data of stator voltage and resolver position coming from the test bench and then uses them as an input for the simulation model. Second, *subscriber 2* is responsible for receiving real-time data of stator current from the physical model to be compared with the result from the simulation. Stator current comparator is used initially for DT tuning in and checking the simulation model accuracy. The resolver decoder block processes the resolver winding signals coming from *subscriber 1*, transferring them into a position value input to the motor block. Measurement block contains scopes to observe motor simulation outputs of electromagnetic torque, angular speed, and the compared stator phase current. Fig.5. shows the simulation model of the EV-PMSM with the two ROS2 subscribers. Fig.6. shows the expanded PMSM block.

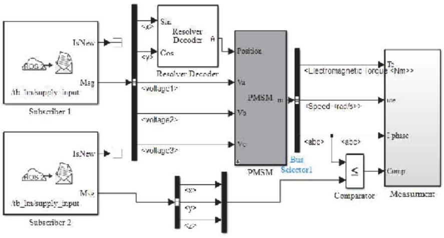


Fig.5. PMSM simulation model.

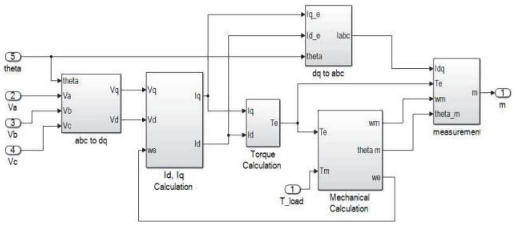
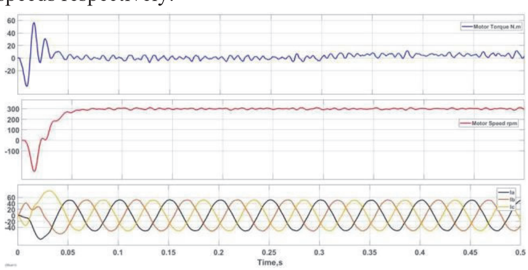


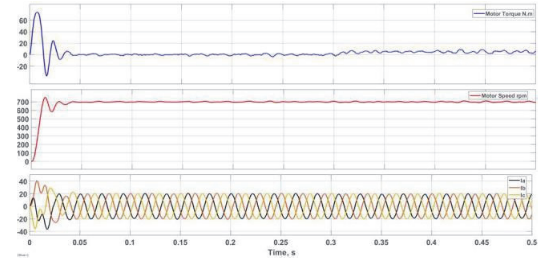
Fig. 6. Expanded PMSM Block.

IV. Results and Discussion

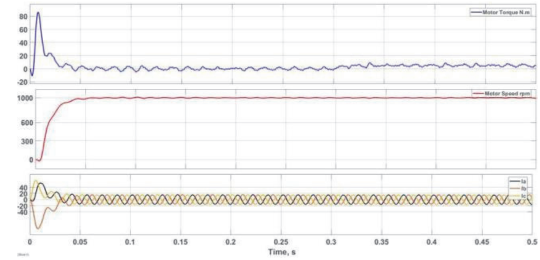
The objective of this research is to demonstrate the effectiveness and readiness of the DT concept for the EV-motor torque estimation. The physical model was run in parallel with the simulation virtual model of the motor under two operating speeds cases in no loading conditions to validate the simulation model's accuracy in real-time. Stator voltage and rotor position were fed to the simulation model in real-time through the ROS2 node. Stator current from the physical model received in subscriber 2 was compared with results from simulation. Motor Torque and speed can be observed through the scopes. Fig. 7 shows the resultant motor electromagnetic torque, speed and, stator current obtained from the simulation model for three operating cases a. low (300 rpm), b. medium (700 rpm), and c. high (1000 rpm) speeds respectively.



a.



b.

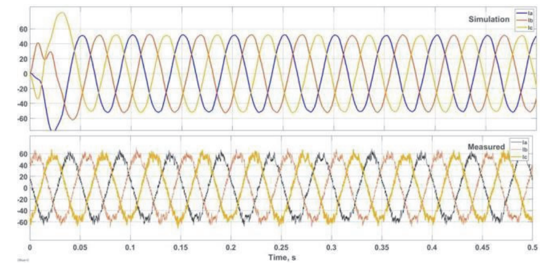


c.

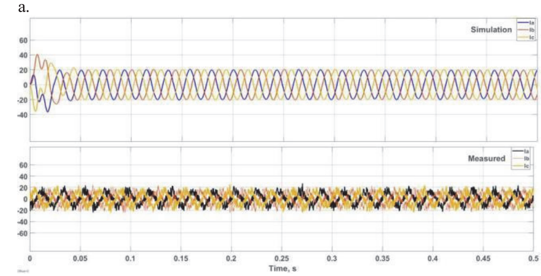
Fig. 7. a, b, c. Estimated (Torque, Speed & stator Current) vs time from simulation model.

The reduction in stator current with the increased speed is noticeable in Fig.7 a, b, c. that resonates with the normal operation of the traction motor profile.

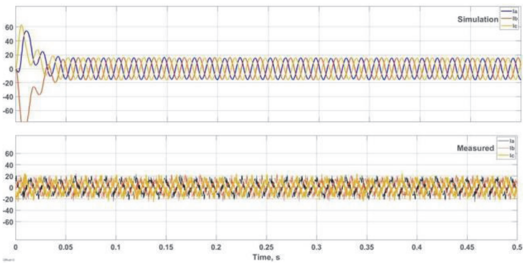
Fig. 8 a, b, c shows the three-phase stator currents obtained from the simulation model and the measured from the test bench for the same three operating speed cases of Fig 7. a, b, c respectively.



a.



b.



c.

Fig. 8. a, b and c. Comparison between simulated and measured stator current vs time – three operating speed cases respectively.

Fig. 8. a, b, and c show high agreement between the measured values of stator phase current obtained from testbench and its counterpart derived from the simulation model, this proves the validity of the proposed simulation model.

There is a noticeable harmonic in the measured stator current waveform compared to that obtained from the simulation at an acceptable level with no fundamental differences due to the noises resulting from sensors.

V. Conclusion

This paper proposes a conceptual methodology for designing a DT of an EV- PMSM. DT is functioned to estimating the motors' electromagnetic torque. The architecture of the motor DT comprises three main components: physical model, Virtual model and, data exchange set (Communication unit). EV-PMSM physical model (test bench) were described in detail. The motor analytical *dq* model and its mathematical equations were presented. MATLAB/Simulink was used to build the motor virtual model based on the proposed analytical model. The concept of DT is achieved by linking both virtual and physical model using data exchange set (communication model) based on ROS2 nodes data messages exchange.

The obtained results of estimated torque from the virtual model are promising and prove the proposed DT effectiveness.

The DT concept proposed in this paper can be developed in the future to formulate a complete DT model through which the physical motor model can be controlled.

Acknowledgment

The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of an autonomous electric vehicle"

References

[1] A. Rassolkin, V. Rjabtsikov, T. Vaimann, A. Kallaste, V. Kuts, and A. Partyshev, “Digital Twin of an Electrical Motor Based on Empirical Performance Model,” Oct. 2020, doi: 10.1109/ICEPDS47235.2020.9249366.

[2] A. Rassõlkin et al., “Implementation of digital twins for electrical energy conversion systems in selected case studies,” Proc. Est. Acad. Sci., vol. 70, no. 1, pp. 19–39, 2021, doi: 10.3176/PROC.2021.1.03.

[3] M. Ibrahim, A. Rassõlkin, T. Vaimann, and A. Kallaste, “Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System,” Sustain. 2022, Vol. 14, Page 601, vol. 14, no. 2, p. 601, Jan. 2022, doi: 10.3390/SU14020601.

[4] N. A. Elsonbaty, M. A. Enany, and M. I. Hassanin, “An Efficient Vector Control Policy for EV-Hybrid Excited Permanent-Magnet Synchronous Motor,” World Electr. Veh. J. 2020, Vol. 11, Page 42, vol. 11, no. 2, p. 42, May 2020, doi: 10.3390/WEVJ11020042.

[5] Y. Guo, Z. Liu, W. Deng, and F. Blaabjerg, “Modeling of electric vehicle driven by PMSM based on torque control,” Proc. 2012 2nd Int. Conf. Instrum. Meas. Comput. Commun. Control. IMCCC 2012, pp. 1020–1024, 2012, doi: 10.1109/IMCCC.2012.241.

[6] B. Forrier, F. Naets, and W. Desmet, “Broadband load torque estimation in mechatronic powertrains using nonlinear kalman filtering.,” IEEE Trans. Ind. Electron., vol. 65, no. 3, pp. 2378–2387, Mar. 2018, doi: 10.1109/TIE.2017.2739709.

[7] W. Nawae and K. Thongpull, “PMSM Torque Estimation Based on Machine Learning Techniques,” Proc. 2020 Int. Conf. Power, Energy Innov. ICPEI 2020, pp. 137–140, Oct. 2020, doi: 10.1109/ICPEI49860.2020.9431433.

[8] Y. Saadi, R. Sehab, A. Chaibet, M. Boukhnifer, and D. Diallo, “Sensorless control of Switched Reluctance Motor with unknown load torque for EV application using Extended Kalman Filter and second order sliding mode observer,” Proc. IEEE Int. Conf. Ind. Technol., vol. 2018-February, pp. 522–528, Apr. 2018, doi: 10.1109/ICIT.2018.8352231.

[9] V. Rjabtsikov et al., “Digital Twin Service Unit for AC Motor Stator Inter-Turn Short Circuit Fault Detection,” Jan. 2021, doi: 10.1109/IWED52055.2021.9376328.

[10] F. Toso, R. Torchio, A. Favato, P. G. Carlet, S. Bolognani, and P. Alotto, “Digital twins as electric motor soft-sensors in the automotive industry,” in 2021 IEEE International Workshop on Metrology for Automotive, MetroAutomotive 2021 - Proceedings, Jul. 2021, pp. 13–18, doi: 10.1109/MetroAutomotive50197.2021.9502885.

[11] H. Brandtstaedter, L. Hübner, A. Jungiewicz, C. Ludwig, E. Tsouchnika, and U. Wever, “DIGITAL TWINS FOR LARGE ELECTRIC DRIVE TRAINS Copyright Material PCIC Europe Paper No. PCIC Europe EUR18_04 Germany Germany Germany,” 2018.

[12] S. Venkatesan, K. Manickavasagam, N. Tengenkai, and N. Vijayalakshmi, “Health monitoring and prognosis of electric vehicle motor using intelligent-digital twin,” IET Electr. Power Appl., vol. 13, no. 9, pp. 1328–1335, Sep. 2019, doi: 10.1049/IET-EPA.2018.5732.

[13] B. Bilgin et al., “Modeling and Analysis of Electric Motors: State-of-the-Art Review,” IEEE Trans. Transp. Electrif., vol. 5, no. 3, pp. 602–617, 2019, doi: 10.1109/TTE.2019.2931123.

[14] J. Liu and W. Chen, “Generalized DQ model of the permanent magnet synchronous motor based on extended park transformation,” 1st Int. Futur. Energy Electron. Conf. IFEEC 2013, pp. 885–890, 2013, doi: 10.1109/IFEEC.2013.6687627.

[15] D. S. More, H. Kallum, and B. G. Fernandes, “D-q equivalent circuit representation of three-phase flux reversal machine with full pitch winding,” PESC Rec. - IEEE Annu. Power Electron. Spec. Conf., pp. 1208–1214, 2008, doi: 10.1109/PESC.2008.4592094.

[16] A. Rassõlkin, R. Sell, and M. Leier, “Development Case Study of the First Estonian Self-Driving Car, ISEAUTO,” Electr. Control Commun. Eng., vol. 14, no. 1, pp. 81–88, 2018, doi: 10.2478/ecce-2018-0009.

Appendix 5

Publication V

Ibrahim, M., Rjabtšikov, V. and Rassõlkin, A. (2025), Digital shadow of an electric vehicle-permanent magnet synchronous motor drive for real-time performance monitoring. Dig Twins and App e12024. https://doi.org/10.1049/dgt2.12024.

Received: 11 October 2024 | Revised: 2 December 2024 | Accepted: 29 December 2024

DOI: 10.1049/dgt2.12024

Digital Twins and Applications

ORIGINAL RESEARCH

Digital shadow of an electric vehicle-permanent magnet synchronous motor drive for real-time performance monitoring

Mahmoud Ibrahim | **Viktor Rjabtšikov** | **Anton Rassõlkin**

Department of Electrical Power Engineering and Mechatronics, Tallinn University of Technology, Tallinn, Estonia

Correspondence

Mahmoud Ibrahim.
Email: mahmoh@taltech.ee

Funding information

Estonian Research Competency Council, Grant/Award Number: PSG453; Eesti Teadusagentuur

Abstract

Digital twin (DT) technology has been utilised in many applications including electric vehicles (EVs). A DT is a virtual representation of a physical object, enabled through real-time data integration, simulation, and optimisation tools. Unlike conventional simulations, which are typically offline and lack real-time interaction, a DT continuously synchronises with the physical system, enabling dynamic performance monitoring and predictive analytics. Achieving a full DT involves progressive stages, with the digital shadow (DS) being the final step before realising a bidirectional DT. Building a DS provides a scalable real-time performance monitoring and fault detection framework, enabling proactive decision-making in EV operations. This study introduces a DS system specifically designed to monitor the performance of a permanent magnet synchronous motor (PMSM) drive system in EVs, marking a critical phase towards a complete DT. The methodology for creating the DS is detailed, including the establishment of a comprehensive test bench for the EV powertrain as the physical reference model. The mathematical model of the EV-PMSM was formulated, and an advanced estimation model utilising the extended Kalman filter (EKF) was implemented. MATLAB/Simulink was employed to develop the motor's digital model. Real-time data acquisition from the physical model was facilitated through a data acquisition system (DAS) equipped with a controller area network (CAN) communication interface. The digital model underwent thorough validation against sensory data collected from the test bench. The motor digital model was deployed to a DS framework enabled through real-time data flow from the actual EV during real-world driving conditions. The results demonstrated a high accuracy of 97% between the DS predictions and the corresponding EV data, confirming the DS's reliability. These findings pave the way for future advancements, including bidirectional interaction and the realisation of a full DT.

KEYWORDS

digital shadow, electrical engineering, modelling, monitoring

1 | INTRODUCTION

The gradual change from internal combustion engines (ICEs) to electric motors has impacted the global automotive industry.[1] Compared to conventional ICEs, electric propulsion systems have a significantly higher power density due to their smaller dimensions and lower weight. Heat loss of electric motors has been reduced to only about 10%, whereas over 90% of the electric energy is converted into mechanical energy.[2] Additionally, electric vehicle (EV) drive systems run at considerably higher rotational speeds, a fact that makes e-mobility test benches a challenge for testing.[3] In addition to increased rotational speed, the dynamic behaviour of electric drives places high requirements on future e-mobility test bench concepts.

With the fourth industrial revolution, digital twin (DT) technology was brought to enhance the development process.

Digital twin is a virtual replica of a physical object that can be used to simulate and analyse its performance in real time.[4] Although simulations and DTs both utilise digital models to replicate the behaviour of physical systems, DTs provide a far more comprehensive and interactive environment for analysis. The key distinction lies in their scope and functionality. Conventional simulations are typically static, focusing on a single process or scenario without real-time data integration. In contrast, a DT incorporates continuous two-way data exchange between the physical system and its virtual counterpart, enabling real-time monitoring, predictive analysis, and system optimisation.[5] DTs offer significant benefits for EV applications including condition monitoring, performance optimisation, predictive maintenance, energy efficiency improvements, virtual testing, fault diagnosis, and component lifetime prediction.[6]

Developing a fully functional DT is a gradual process that progresses through several stages, each adding more complexity and functionality. The process typically begins with the creation of an advanced digital model, which serves as a virtual representation of the physical system but operates without real-time data integration. As the development progresses, the model transitions into a digital shadow (DS), where real-time data from the physical system is streamed to its virtual counterpart. This unidirectional data flow enables the virtual model to accurately reflect the physical system's performance and simulate real-time conditions within the virtual environment. In contrast, a true DT enables bidirectional data flow, allowing the virtual model not only to monitor and analyse the physical system but also to influence and optimise its behaviour in real time, driving improvements in efficiency, reliability, and decision-making.[7] Figure 1 illustrates the key difference between digital model, DS, and DT.

Despite the term DT being widely used in research and industry, in many cases, the actual implementations often represent DS models rather than DTs. The key distinction lies in the lack of bidirectional communication between the physical system and its virtual counterpart. In many instances, the data flows only from the physical system to the virtual model, without the ability of the virtual model to influence or control the physical object. This one-way data exchange is a hallmark of a DS, not an actual DT, where the virtual model would be able to adjust or interact with the physical system dynamically.

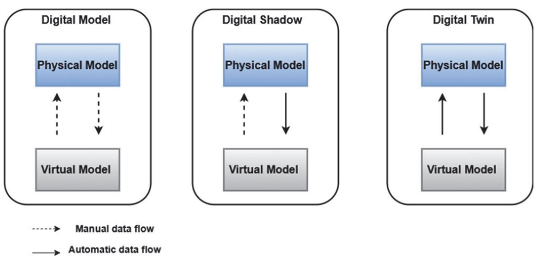


FIGURE 1 Comparison between DM, digital shadow (DS), and digital twin (DT).

This issue has been widely overlooked in the literature, where the term DT is often used generically to describe a real-time virtual representation of a physical system, even in cases where only a DS has been achieved. This misrepresentation can lead to confusion in understanding the actual capabilities of the models being discussed.[8]

DS plays a critical role in developing and optimising complex systems, particularly during the transitional phase towards full DT integration. One of the primary benefits of a DS is its ability to provide real-time insights into the operation of a physical system, without requiring intervention or control over the physical object. This capability is crucial for applications like monitoring, diagnostics, and lifetime prediction, where continuous data collection enables tracking performance metrics, identifying potential failures, and optimising performance without disrupting operations. Additionally, DS is essential for historical data analysis, which is invaluable for improving system designs and informing future innovations.[9]

The concept of DS and DT has been explored extensively in the context of EV motor drive systems. For instance, Venkatesan et al.[10] proposed an intelligent DT for health monitoring and prognosis of an EV permanent magnet synchronous motor (PMSM), utilising artificial neural networks and fuzzy logic to estimate remaining useful life and critical parameters, with both in-house and remote monitoring capabilities validated through simulation results. Rjabtsikov et al.[11] developed a communication framework using a robotics operating system for a DT model of an induction motor dedicated to condition monitoring. Hu et al.[12] introduced a DT for induction motor health monitoring utilising Ramanujan periodic transform and Bayesian-updated calibration to detect fault signatures with high fidelity, overcoming challenges like noise interference and varying working conditions. A predictive maintenance tool for induction motors was developed using DT and industrial Internet of things concepts, combining sensor-based current and temperature monitoring with a high-fidelity finite element method (FEM) model to provide accurate diagnostics and maintenance predictions.[13] Toso et al.[14] deployed a DS of an EV-induction motor, serving as a soft sensor for condition monitoring of torque and internal motor temperature. An extended Kalman filter (EKF)-based DT was proposed to accurately estimate the states of a speed sensorless rotor field-oriented controlled induction motor driven.[15] Adamou and Chakib[16] proposed a hybrid DS model using data-driven and physics-based approaches, combined with an adaptive neuro-fuzzy inference system, to estimate the real-time energy efficiency and losses of induction motors.

Another essential aspect of DS used for performance monitoring is the integration of soft sensors. A soft sensor is a type of software that, given the available information, processes what a physical sensor otherwise would. It learns to interpret the relationships between the different variables and observes readings from the different instruments.[14] The idea behind it is that if the simulation has been run under accurate inputs, and it mimics the behaviour of a product in real time, then it applies to the instrument that the simulation model by taking measurements at different locations. Readings from the virtual sensor, which is where measurements are been taken, can be used to complement physical sensors. The main advantage of this is that virtual sensors can be placed on a virtual model anywhere which is way different from the real physical sensors.[17] Overall, the use of virtual sensors in EV motors offers cost savings, increased reliability, flexibility, real-time monitoring, and performance optimisation. By relying on advanced algorithms and existing data sources, soft sensors provide valuable insights and measurements, contributing to the efficient and reliable operation of EV powertrains. From the above literature, it can be deduced that much of the EV performance monitoring research has focused on post-failure analysis or reactive maintenance approaches. This gap, in which real time proactive monitoring was limited, is addressed through the integration of DS technology. A DS introduces a real-time layer of operational monitoring that goes beyond just analysing data after issues occur. This paper introduces the procedures for developing a DS of an EV powertrain for real-time performance monitoring as a precursor to a fully realised DT. The paper is organised as follows: an overview of DT technology in EV applications is provided in Section 1. Section 2 outlines the development procedures for the PMSM digital model, starting with the physical system description in Section 2.1, which introduces the physical motor under study and its test bench. Section 2.2 derived the advanced d–q model of the PMSM, while Section 2.3 presents the mathematical model of the coupled transmission unit. Section 2.4 details the EKF estimation model, and Section 2.5 describes the developed motor digital model. Section 2.6 introduces the data exchange unit that facilitates real-time data flow between the physical and digital counterparts. Section 3 highlights the validation and tuning process of the developed motor digital model. Section 4 addresses the deployment of the model to the DS framework enabled through the actual EV during real-world driving conditions. Finally, section 5 concludes the paper by summarising the key findings and outlining future research directions.

2 | ELECTRIC VEHICLE PERMANENT MAGNET SYNCHRONOUS MOTOR DIGITAL MODELLING

The process of developing a DS typically involves several key phases: model identification, modelling, validation and tuning, and deployment to a real-time framework. Model identification involves defining the main parameters that qualify the modelling process. The next phase, modelling, includes the creation of a digital model of the physical system. This digital model is tuned to closely simulate the performance of the system. The digital model is then validated and fine-tuned by comparing its performance to the real measurements from the physical system. Finally, model deployment to the DS framework enabling real-time data flow from the physical counterpart to the digital model.

2.1 | Physical system

The motor drive physical model was taken from ISEAUTO, an Estonian self-driving vehicle that was designed as a last-mile vehicle for use primarily on the Tallinn University of Technology (TalTech) campus, with a maximum speed of 20 km/h which equal to 30% of motor rated speed.[18] It was built based on a Mitsubishi i-mev trolly Y4F1 PMSM. Table 1 illustrates the PMSM parameters.

To obtain a dependable physical model of the EV powertrain, a PC-controlled test bed was created to replicate the EV traction drive system. It was designed as follows: the electric vehicle-permanent magnet synchronous motor (EV-PMSM) is driven by a 55-kW heavy-duty traction inverter unit with a vector control strategy. The EV inverter is controlled by a visual interface PC unit. The battery emulator unit is combined with a regenerative AC to DC converter set to behave like the real batteries of an EV. The motor is equipped with an internal resolver unit for position measurement. The motor shaft is driving a (F1E1A) four-gear double-stage type transmission unit, which is coupled with a differential unit with an overall reduction ratio of 0.66. Two loading emulator drives are mounted on both differential sides to replicate the vehicle's driving wheels. The loading emulators are controlled through a PC interface unit with the ability to define the percentage load torque applied. A modular power analyser and data acquisition system (DAS) is used to acquire different sensory data from the test bench. Figure 2 shows the test bench of the EV motor drive system.

2.2 | Electric vehicle-permanent magnet synchronous motor mathematical model

Modelling and analysing electric motors in EV applications can be accomplished in several ways. The simulation time and the accuracy of the results may vary significantly depending on the

TABLE 1 Under study motor parameters.

Parameter	Description	Value	Unit
P	Number of pole pairs	4	-
Rs	Stator resistance	0.10087	Ω
L_q	q-axis inductance	$2.4{\cdot}10^{-4}$	H
L_d	d-axis inductance	$1.9{\cdot}10^{-4}$	H
λ_{pm}	Permanent magnet linkage flux	0.087	Wb
N_r	Rated speed	3000	rpm
$P_{\max}$	Maximum output power	47	kW
T_r	Maximum torque	180	N·m
J	Motor inertia	0.0001	kg·m^2
b	Damping coefficient	0.0001	N·m·s/rad

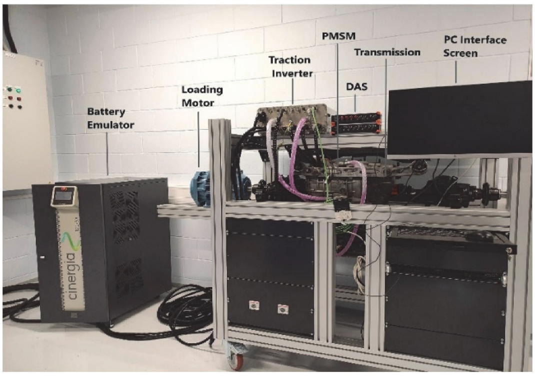


FIGURE 2 Permanent magnet synchronous motor (PMSM) drive test bench.

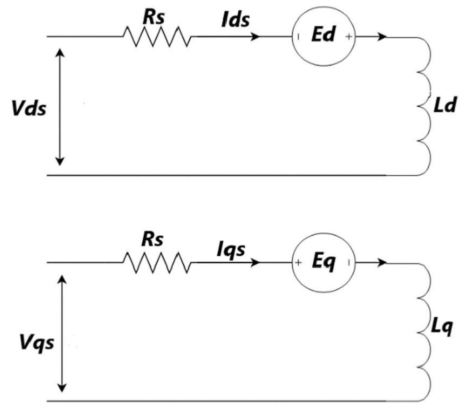


FIGURE 3 Permanent magnet synchronous motor (PMSM) d–q equivalent circuit.

selected analysis technique. The three main methods for modelling an electric motor, and studying its various variables, are as follows: finite element modelling (FEM), d–q equivalent circuit (d–q), and magnetic equivalent circuit.[19]

Among these, the d–q equivalent circuit model is widely used due to its balance of computational efficiency and accuracy. This model simplifies the mathematical representation of the motor by transforming the three-phase system into a two-axis coordinate system (d and q axes), capturing the fundamental dynamics required for calculating key variables such as voltage, current, and electromagnetic torque. Its simplicity and speed make it particularly suitable for real-time applications, especially in frameworks like the DS, where continuous monitoring and real-time data processing are essential. Whereas the d–q model is effective for many practical applications, it does have limitations that arise under certain operating conditions. At high speeds, cross-coupling between the d and q axes becomes more significant, leading to deviations from the model's basic assumptions, however the d–q model remains a reliable and practical choice for the operating conditions of the EV motor under study.[20] Designed for urban mobility, the EV operates at a maximum speed of 20 km/h, approximately 30% of the motor's rated speed. In this range, the likelihood of significant non-linearities, such as magnetic saturation or high-frequency losses, is minimal. This makes the d–q model highly suitable for real-time performance monitoring and within the DS framework. The d- and q-axis equivalent circuit of the PMSM is shown in Figure 3.

- Stator reference frame equations

$$V_a = R_s i_a + \frac{d\lambda_a}{dt} \tag{1}$$

$$V_b = R_s\, i_b + \frac{\mathrm{d}\lambda_\mathrm{b}}{\mathrm{dt}} \tag{2}$$

$$V_c = R_s\, i_c + \frac{\mathrm{d}\lambda_\mathrm{c}}{\mathrm{dt}} \tag{3}$$

where (V_a, V_b, V_c) are the phase voltages , (i_a, i_b, i_c) Are the phase currents, and $(\lambda_a, \lambda_b, \lambda_c)$ are the flux linkages.

- The flux linkages are defined as follows:

$$\lambda_a = L_s i_a + L_m\left(i_{m_a} + i_{m_b} + i_{m_c}\right) \tag{4}$$

$$\lambda_b = L_s i_b + L_m\left(i_{m_a} + i_{m_b} + i_{m_c}\right) \tag{5}$$

$$\lambda_c = L_s i_c + L_m\left(i_{m_a} + i_{m_b} + i_{m_c}\right) \tag{6}$$

where (L_s) is the stator self-inductance, (L_m) is the mutual inductance, and $\left(i_{m_a}, i_{m_b}, i_{m_c}\right)$ are the magnetising currents.

- Rotor d–q reference frame:

The transformation from the abc to dq0 reference frame is performed using the Park transformation:

$$V_d = \frac{2}{3}\left(V_a \cos\theta + V_b \cos\left(\theta - \frac{2\pi}{3}\right) + V_c \cos\left(\theta + \frac{2\pi}{3}\right)\right), \tag{7}$$

$$V_q = \frac{2}{3}\left(-V_a \sin\theta - V_b \sin\left(\theta - \frac{2\pi}{3}\right) - V_c \sin\left(\theta + \frac{2\pi}{3}\right)\right), \tag{8}$$

$$V_0 = \frac{1}{3}(V_a + V_b + V_c), \tag{9}$$

and

$$i_d = \frac{2}{3}\left(i_a \cos\theta + i_b \cos\left(\theta - \frac{2\pi}{3}\right) + i_c \cos\left(\theta + \frac{2\pi}{3}\right)\right), \tag{10}$$

$$i_q = \frac{2}{3}\left(-i_a \sin\theta - i_b \sin\left(\theta - \frac{2\pi}{3}\right) - i_c \sin\left(\theta + \frac{2\pi}{3}\right)\right), \tag{11}$$

$$i_0 = \frac{1}{3}(i_a + i_b + i_c), \tag{12}$$

- Dynamic equations in dq0 Frame

The dynamic equations in the dq0 frame, considering both the stator and rotor reference frames, are given as follows:

$$V_d = R_s i_d + \frac{d\lambda_d}{dt} - \omega_e \lambda_q \tag{13}$$

$$V_q = R_s i_q + \frac{d\lambda_q}{dt} + \omega_e \lambda_d \tag{14}$$

$$V_0 = R_s i_0 + \frac{d\lambda_0}{dt} \tag{15}$$

where:

$$\lambda_d = L_d i_d + \lambda_m, \tag{16}$$

$$\lambda_q = L_q i_q, \tag{17}$$

$$\lambda_0 = L_0 i_0. \tag{18}$$

Here, λ_m is the permanent magnet flux linkage, λ_d λ_q are the d-axis and q-axis inductances, respectively, and L_0 is the zero-sequence inductance.

Electromagnetic torque

The electromagnetic torque (T_e) generated by the PMSM can be expressed as follows:

$$T_e = \frac{3}{2}p(\lambda_d i_q - \lambda_q i_d) = \frac{3}{2}p\left[\lambda_m i_q + (L_d - L_q) i_d i_q\right], \tag{19}$$

where p is the number of pole pairs.

Mechanical dnamics

The mechanical dynamics of the rotor are governed using the following equation:

$$J\frac{d\omega_m}{dt} = T_e - T_L - B\omega_m, \tag{20}$$

where:

- J is the rotor's moment of inertia,
- B is the viscous friction coefficient,
- ω_m is the mechanical angular velocity,
- T_L is the load torque.

2.3 | Transmission unit mathematical model

The transmission unit represents a double-stage gearbox with four gears. The following equations represent the final speed and torque on the vehicle's front wheels.

Torque equation between gears A and B:

$$T_a = \left(T_\beta \times \left(N_a \,/\, N_\beta\right) \times K_1\right) \tag{21}$$

Torque equation between gears B and C:

$$T_\beta = \left(T_\varrho \times \left(N_\beta \,/\, N_\varrho\right) \times K_2\right) \tag{22}$$

Torque equation between gears C and D:

$$T_\varrho = \left(T_\varphi \times \left(N_\varrho \,/\, N_\varphi\right) \times K_3\right) \tag{23}$$

where: K_1, K_2, and K_3 represent the stiffness factors for gears A–B, B–C, and C–D, respectively.

The modified speed equations, considering lubricant viscosity, become:

Speed equation between gears A and B:

$$\omega_a = \left(\omega_\beta \times \left(N_\beta \,/\, N_a\right)\right) \,/\, \left(\mu_1\right) \tag{24}$$

Speed equation between gears B and C:

$$\omega_\beta = \left(\omega_\varrho \times \left(N_\varrho \,/\, N_\beta\right)\right) \,/\, \left(\eta_2 \times \mu_2\right) \tag{25}$$

Speed equation between gears C and D:

$$\omega_\varrho = \left(\omega_\varphi \times \left(N_\varphi \,/\, N_\varrho\right)\right) \,/\, \left(\eta_3 \times \mu_3\right) \tag{26}$$

where: μ_1, μ_2, and μ_3 represent the lubricant viscosity factors for gears A–B, B–C, and C–D, respectively.

2.4 | Extended Kalman filter estimation model

In the context of DS modelling, the EKF emerges as a crucial estimation tool, enabling accurate torque and speed estimation, which serves as a soft sensor in this framework. The EKF is well suited for handling the non-linear nature of PMSM dynamics, where magnetic fields and mechanical motion are intricately coupled. As the DS relies on real-time data from the physical system without bidirectional interaction, the EKF plays a vital role in processing this incoming data to estimate key parameters such as torque. The EKF operates in two distinct phases: a prediction phase, where the motor's mathematical model is used to forecast the next system state (e.g. estimated torque), and an update phase, where this prediction is adjusted based on real-time measurements from the physical system. This dual-phase approach allows the EKF to deliver reliable torque estimates, even in the presence of noisy sensor data and system uncertainties, making it an indispensable component in the DS for condition monitoring and performance evaluation. The basics of the EKF can be mathematically represented as follows:

- State definition

State vector (x)

$$x = \begin{bmatrix} \lambda_d \\ \lambda_q \\ \omega_e \end{bmatrix} \tag{27}$$

Input vector (u)

$$u = \begin{bmatrix} V_{ds} \\ V_{qs} \end{bmatrix} \tag{28}$$

Output vector (y)

$$y = \begin{bmatrix} I_{ds} \\ I_{qs} \end{bmatrix} \tag{29}$$

- State transition

Non-linear dynamics

$$x = f(x,u) = \begin{bmatrix} -\frac{R_s}{L_d}\lambda_d + \omega_e\,\lambda_q + \frac{V_{ds}}{L_d} \\ -\omega_e\,\lambda_q - \frac{R_s}{L_q} - \lambda_d \\ -b\omega_e + \frac{3}{2} * \frac{P}{J}\,\left(\lambda_d I_{qs} - \lambda_q I_{ds}\right) \end{bmatrix} \tag{30}$$

Jacobian state function (F)

$$F = \frac{df}{dx} = \begin{bmatrix} -\frac{R_s}{L_d} & \omega_e & \lambda_q \\ -\omega_e & -\frac{R_s}{L_q} & -\lambda_d \\ \frac{3}{2}\frac{P}{J}I_{qs} & -\frac{3}{2}\frac{P}{J}I_{ds} & -b \end{bmatrix} \tag{31}$$

- Measurement (observation model)

Output equation

$$Y = h(x) = \begin{bmatrix} \frac{\lambda_d - \lambda_{pm}}{L_d} \\ \frac{\lambda_q}{L_q} \end{bmatrix} \tag{32}$$

Jacobian measurement function (H)

$$H = \frac{dh}{dx} = \begin{bmatrix} \frac{1}{L_d} & 0 & 0 \\ 0 & \frac{1}{L_q} & 0 \end{bmatrix} \tag{33}$$

Covariance matrix (p)

$$P = \begin{bmatrix} P_{\lambda_{d\lambda_d}} & P_{\lambda_{d\lambda_q}} & P_{\lambda_{d\omega_e}} \\ P_{\lambda_{q\lambda_d}} & P_{\lambda_{q\lambda_q}} & P_{\lambda_{q\omega_e}} \\ P_{\omega_{e\lambda_d}} & P_{\omega_{e\lambda_d}} & P_{\omega_e\omega_e} \end{bmatrix} \tag{34}$$

Where diagonal elements of the matrix represent the variance of each state variable, indicating the degree of uncertainty or spread in their estimates.

Identity matrix (I)

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{35}$$

EKM initialisation

Initialise the state estimate x_0 and covariance P_0.

Where p is the initial prediction covariance.

- Prediction procedures

Predict the next state: $x^- = f(x,u)$ using the state transition model.

Linearise the state transition around x to compute the Jacobian matrix F.

Predict the covariance:

$$P^- = FPF^T + Q \tag{36}$$

Where Q: process noise covariance.

Update

Compute the Kalman gain

$$K = P^- H^T \left(HP - H^T + R\right)^{-1} \tag{37}$$

Update the state estimate

$$x = x^- + K(y - h(x^-)) \tag{38}$$

Update the covariance

$$P = (I - KH)P^{-} \quad (39)$$

2.5 | Permanent magnet synchronous motor digital model

The motor digital model was developed using MATLAB/Simulink platform, based on the mathematical model derived in the previous subsection B. This platform was chosen for its robust real-time simulation capabilities, extensive availability, and support for various communication frameworks, making it a sufficient tool for developing the motor DS. The digital model was designed to receive real-time inputs, including stator voltage, current, and torque demand, and incorporates a PMSM model that captures the motor's dynamic behaviour. The EKF was integrated through an observer block, enabling the system to estimate motor torque and speed based on the consumed current and applied voltage. Additionally, the input voltage carries the required frequency, which directly influences the motor speed, ensuring the model accurately reflects the relationship between electrical inputs and mechanical outputs. This integration allows the EKF to better handle the non-linear dynamics of the PMSM, where magnetic fields and mechanical motion are intricately coupled. The virtual model also includes a transmission block that replicates the physical transmission unit, and it was designed as a reduced-order model with a fixed solver, enabling it to process real-time data promptly and accurately for condition monitoring. To facilitate real-time interaction, the model features two controller area network (CAN) receiving blocks for capturing input voltage and consumed current, ensuring that the virtual model's outputs remain synchronised with those of the physical system. This real-time data exchange ensures the model can perform reliable condition monitoring and performance evaluation. Figure 4 shows the digital model of the EV motor system while Figure 5 shows the expanded PMSM block.

2.6 | Data exchange (communication unit)

A data exchange unit is a software component that facilitates real time one-way data transfer between the physical and virtual models of a DS. Data acquisition system is designed to collect and transmit real-time data from the EV's motor and other components. It includes a CAN bus, which serves as the primary communication protocol for gathering data. The CAN bus efficiently collects this data from various sensors within the testbench, ensuring that the information is accurate and up-to-date. Once the data is gathered by the CAN bus, it is transmitted to a Texas Instrument microcontroller that acts as a middleware. This microcontroller processes the incoming CAN data, serving as an interface between the CAN network and the host model. The microcontroller converts the CAN data into a format that the host model can interpret and ensures that the data is relayed without delay. The host model then receives the processed data from the microcontroller and transmits the data to the target model, which represents the DS of the EV motor system. This entire process occurs in real time, allowing the DS to continuously update with the latest information from the physical system. Figure 6 illustrates the DS architecture.

3 | VALIDATION AND TUNING

The second phase of DS development focused on validation, which ensures that the digital model accurately mirrors the physical system's behaviour. To achieve this, test conditions were carefully selected to cover various modes of the EV driving cycle, reflecting real-world operational scenarios. This approach aimed to validate the virtual model under conditions that simulate different driving situations, such as acceleration, cruising, and deceleration, aligning the tests with actual EV operation. Since the vehicle's maximum speed is 20 km/h, which is equivalent to 1000 rpm for the motor, the drive system was configured to meet this requirement. Several tests were conducted on the motor test bench under various loading and speed conditions as shown in Table 2.

Data acquisition system was used to measure the motor's stator current, stator voltage, and rotor speed during these tests. The measured data from the test cases were saved in the form of MATLAB data files, which were used as inputs for the motor's digital model. During the validation of the digital model, the simulated values of motor speed were compared to the real-world measurements obtained from the physical system under the same test conditions. Motor speed was chosen as the primary variable for validation and tuning because a physical sensor (resolver) was available to measure the actual speed during testing. The results of these comparisons demonstrated that the digital model accurately replicated the behaviour of the physical motor. Figure 7a, b, c, and d shows the validation results from tests A, B, C, and D, respectively.

From the above, several observations can be made regarding the comparison between the digital model's values and the actual measured values. The speed values from the digital model exhibited fewer ripples compared to the actual measured speed, primarily because the model is based on an idealised mathematical representation that does not account for real-world factors such as mechanical vibrations, load variations, and sensor noise. The percentage error between the model's speed and the actual measurements was generally below 5% across all test cases, except for case D, where the error occasionally reached 10% due to the higher torque operation, making the system more sensitive to small variations in model parameters like resistance, inductance, and back-EMF. In contrast, at lower torques (higher speeds), the motor's sensitivity to these parameter changes decreases, leading to a closer match between the digital model and actual speed

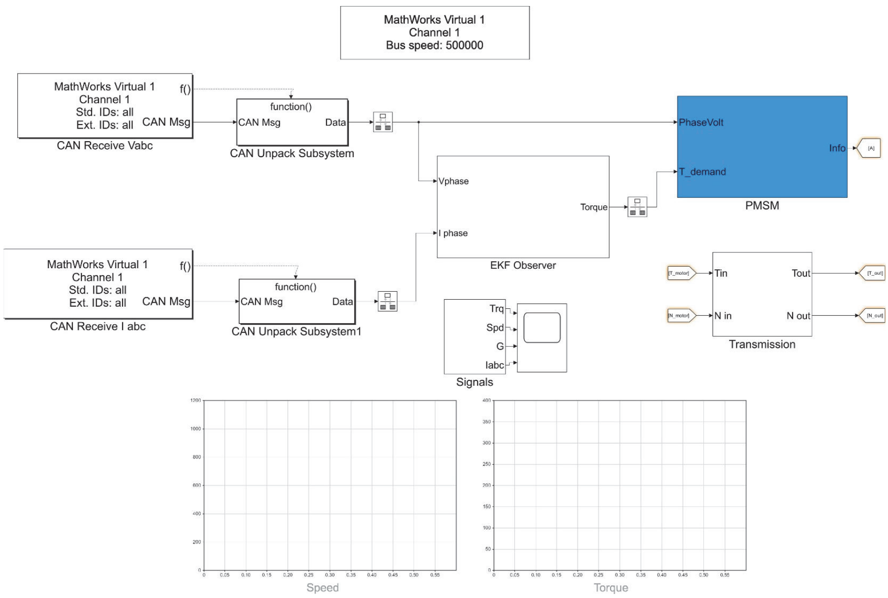


FIGURE 4 Electric vehicle-permanent magnet synchronous motor (EV-PMSM) digital model.

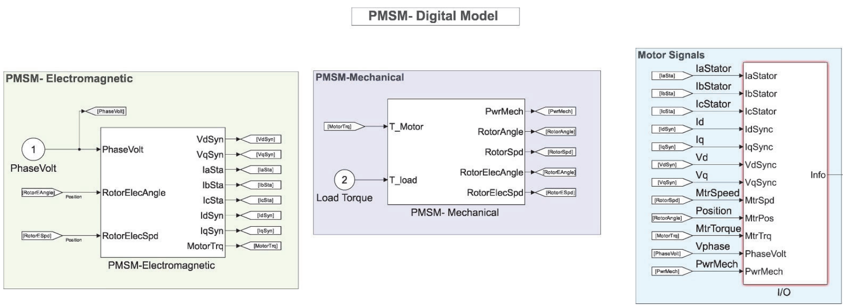


FIGURE 5 Permanent magnet synchronous motor (PMSM) expanded block.

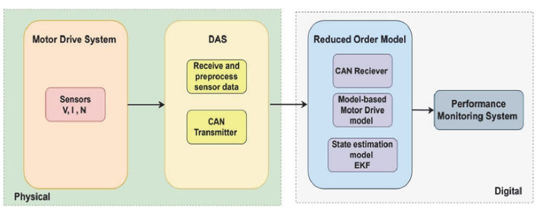


FIGURE 6 Digital shadow (DS) architecture.

TABLE 2 Validation tests conditions.

Case	% Torque demand	Speed (rpm)
A	80%	300
B	60%	500
C	40%	700
D	20%	1000

values. Regarding current, the digital model showed higher accuracy at lower speeds (case) compared to higher speeds (cases A and B) because the motor has more time to adjust the current to the desired level at lower speeds, allowing for greater accuracy. At higher speeds, the reduced time for adjustment and increased vibrations and noise affect current measurement accuracy. While the digital model's current signals displayed more distortion and harmonics, likely due to numerical errors from the limited precision of the simulation, the root mean square values of the steady-state stator currents closely matched the real testbench values, confirming the model's overall validity.

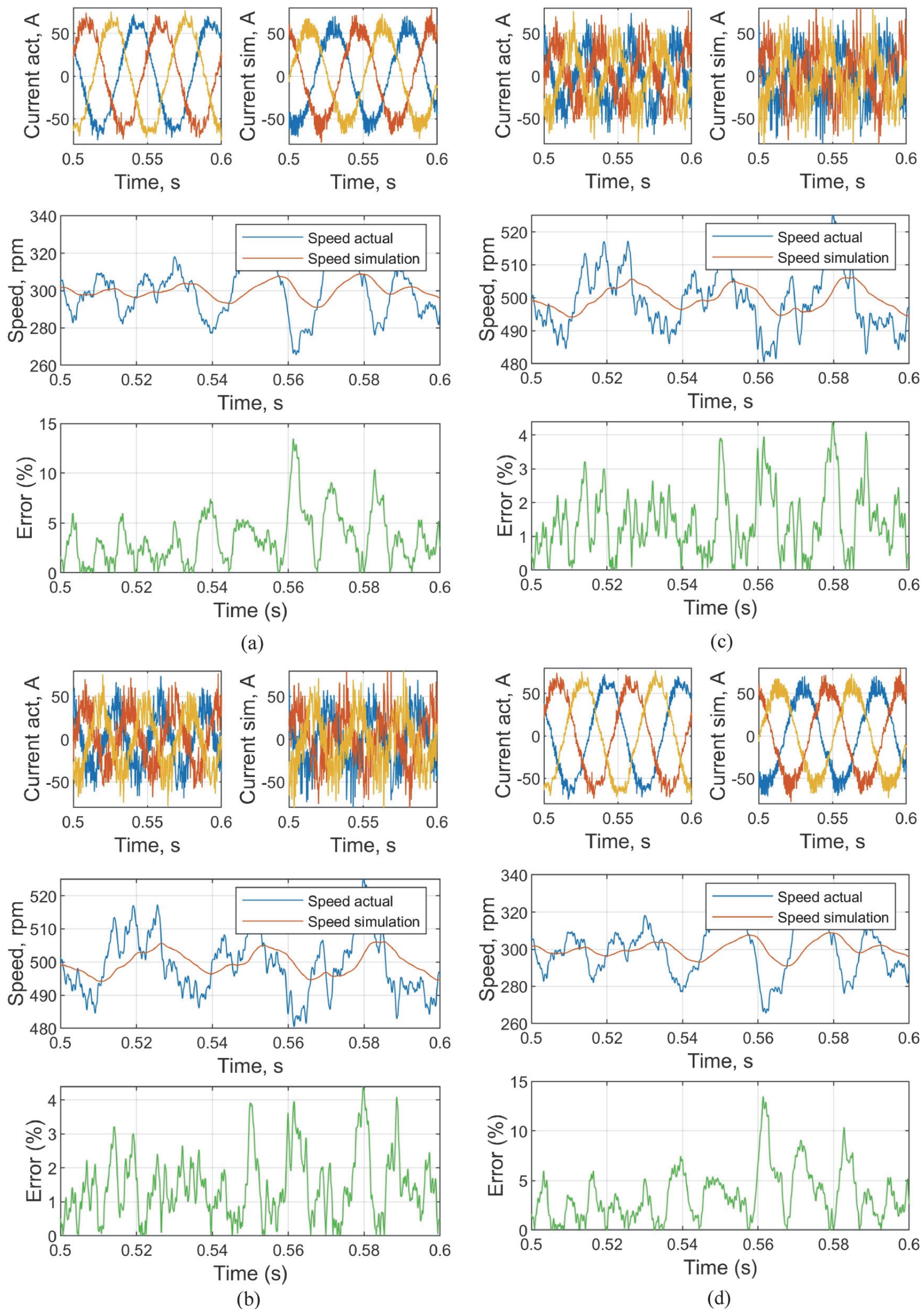


FIGURE 7 (a), (b), (c), and (d) Motor speed and stator current actual versus digital model.

4 | DEPLOYMENT TO DIGITAL SHADOW FRAMEWORK

The final step in the DS development process is deploying the digital model to the DS framework allowing real-time data from the actual EV, ISEAUTO. This phase involves setting up the necessary connection protocols between the physical vehicle and the virtual model, ensuring seamless data exchange from the physical system to the DS. The key objective is to monitor and analyse the vehicle's real-time performance through the DS, which mirrors the behaviour of the physical system based on the incoming data.

ISEAUTO is equipped with a dual GPS antenna system that tracks essential performance metrics such as velocity, distance, and altitude. This GPS allows for a comprehensive assessment of the vehicle's real-world behaviour, providing accurate data that the DS can use to replicate the vehicle's movements and performance. The continuous data stream enables the DS to evaluate various operational parameters, ensuring a close match with the physical vehicle. To enable DS, parallel operations are essential, where both the physical and digital models run simultaneously. For this purpose, the ISEAUTO's motor was equipped with voltage and current sensors directly connected to the DAS hardware unit. The DAS plays a central role, collecting real-time data from these sensors as well as GPS data regarding the vehicle's altitude and velocity. This collected data is then transmitted in real time to the motor's virtual model, which operates concurrently with the physical motor. By ensuring direct connectivity between the physical motor, sensors, DAS, and the digital model, this setup enables continuous data flow and simulated feedback. This parallel operation facilitates the comparison of real-world performance with the DS predictions.

A test area was chosen with a 200-m distance and with a minimum slope at the university campus. Figure 8 shows the GPS data of velocity and altitude from the vehicle's GPS antenna while Figure 9 shows the sensory data of the real-time motor voltage and current collected during the test.

The vehicle started from a standstill at low velocity and then increased gradually till reaching an almost constant velocity. The altitude slightly increased gradually from 22.4 to 23.2 m above sea level which means a slight increase in the loading torque on the motor. The vehicle velocity can be converted to rpm value according to the following equation.

$$v = 0.1885 \cdot N \cdot D \tag{44}$$

*w*here:
v–wheel velocity in km/h,
N–vehicle speed in rpm, and
D–wheel diameter in m.

From the DS side, it can monitor the motor performance in real time during the vehicle test. It outputs the motor's torque and speed as shown in Figure 10 while Figure 11 compares the obtained velocity from the DS with the real one from GPS.

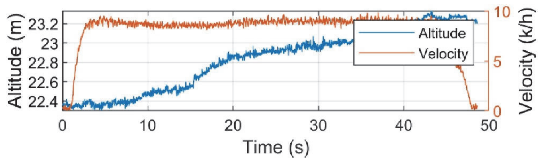


FIGURE 8 ISEAUTO GPS sensor data of vehicle velocity and altitude.

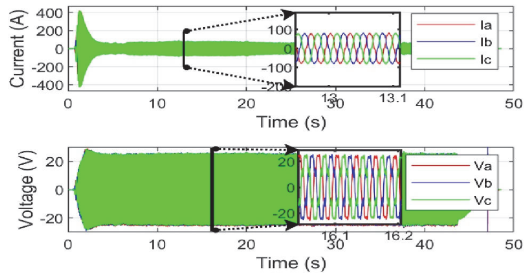


FIGURE 9 ISEAUTO sensor data of motor voltage and current during test.

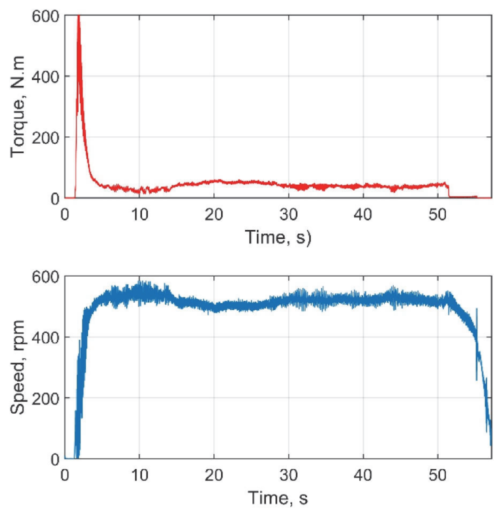


FIGURE 10 Electric vehicle (EV) motor torque and speed from the Digital shadow (DS) model.

The analysis of Figure 10 demonstrates a strong correlation between the motor's torque and current behaviour, as highlighted in Figure 9. The data indicates that the torque needed to move the vehicle from a standstill is significantly high, supported by the corresponding surge in starting current shown in Figure 9. Between seconds 10 and 23, the motor torque increases while there is a slight decrease in speed, indicating acceleration under load, which aligns with the slight rise in altitude observed in Figure 8. From seconds 23–30, the

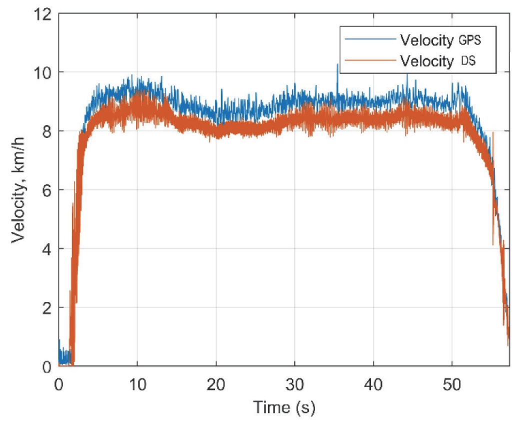


FIGURE 11 Vehicle velocity from GPS and Digital shadow (DS).

speed gradually increases to the nominal test speed as the torque returns to its normal level, showcasing the motor's ability to adapt to changing driving conditions. Figure 11 demonstrates a high level of agreement between the GPS and DS velocities, with a percentage error of less than 3%. This low error rate confirms the accuracy of the DS in monitoring system performance.

The error margin of under 3% reveals 97% DS accuracy, which can be explained by the inherent limitations of GPS accuracy. Factors such as the number of satellites in view, signal quality, and the type of GPS receiver used can all influence the precision of GPS measurements.

The developed DS successfully monitored the motor performance within the defined testing environment, however, its performance outside these boundaries remains untested. Urban environments present different challenges, such as unpredictable traffic patterns, diverse road conditions, and interactions with various vehicles and pedestrians.

5 | CONCLUSION

This paper outlined the development procedures for a DS of an EV-PMSM for condition monitoring, demonstrating the synergy between advanced simulation techniques and real-world sensing. The DS was built on three key components: the physical model (test bench), the digital model, and the real-time data flow between them. The procedures involved data gathering, modelling, validation, and deployment. An advanced analytical d–q model was used as the foundation to develop the motor's virtual model, with an EKF algorithm integrated for torque and speed estimation, functioning as a soft sensor. The developed DS model was validated against the real test bench data, achieving high accuracy in replicating real-world motor behaviours. The DS was tested on the real EV motor system by allowing real-time data flow from onboard sensors through the CAN communication integrated into the DAS during the driving test. The test resulted in an accuracy of 97% in monitoring the motor speed and torque. The use of the DS for condition monitoring of motor torque and speed proved to be an efficient and cost-effective solution to physical sensor problems. By bypassing common challenges such as high installation and maintenance costs, as well as the wear and environmental susceptibility of physical sensors, the DS offers a more reliable and sustainable solution. The model exhibited high speed and accuracy in real-time monitoring, significantly reducing the complexity and cost associated with traditional sensor systems.

AUTHOR CONTRIBUTIONS

Mahmoud Ibrahim: Conceptualisation; investigation; methodology; validation; writing–original draft; formal analysis; writing–original draft. **Viktor Rjabtšikov**: Data curation; investigation; validation; visualisation. **Anton Rassõlkin**: Project administration; resources; supervision; funding acquisition; writing–review & editing.

ACKNOWLEDGEMENTS

The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of autonomous EV".

CONFLICT OF INTEREST STATEMENT

The authors have no conflicts of interest to declare.

DATA AVAILABILITY STATEMENT

Research data are not shared.

ORCID

Mahmoud Ibrahim https://orcid.org/0009-0001-2477-3062

REFERENCES

1. Sanguesa, Julio A., Vicente Torres-Sanz, Piedad Garrido, Francisco J. Martinez, and Johann M. Marquez-Barja. 2021. "A Review on Electric Vehicles: Technologies and Challenges." *Smart Cities* 4(1): 372–404: 2021. https://doi.org/10.3390/SMARTCITIES4010022.
2. Albatayneh, Aiman, Adel Juaidi, Mustafa Jaradat, and Francisco Manzano-Agugliaro. 2023. "Future of Electric and Hydrogen Cars and Trucks: An Overview." *Energies* 16(7): 3230: 2023, Vol. 16, Page 3230. https://doi.org/10.3390/EN16073230.
3. Ibrahim, Mahmoud, Anton Rassõlkin, Toomas Vaimann, and Ants Kallaste. 2024. "Propulsion Drives and Control Algorithms of Electrical Vehicles." *Handbook of Power Electronics in Autonomous and Electric Vehicles*: 139–59. https://doi.org/10.1016/B978-0-323-99545-0.00009-9.
4. Sharma, Angira, Edward Kosasih, Jie Zhang, Alexandra Brintrup, and Anisoara Calinescu. 2022. "Digital Twins: State of the Art Theory and Practice, Challenges, and Open Research Questions." *Journal of Industrial Information Integration* 30: 100383. https://doi.org/10.1016/J.JII.2022.100383.
5. Ibrahim, Mahmoud, Anton Rassõlkin, Toomas Vaimann, and Ants Kallaste. 2022. "Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System." *Sustainability* 14(2): 601: 2022, Vol. 14, Page 601. https://doi.org/10.3390/SU14020601.
6. Li, Xiaokang, Wenxu Niu, and Haobin Tian. 2024. "Application of Digital Twin in Electric Vehicle Powertrain: A Review." *World Electric Vehicle Journal* 15(5): 208: 2024, Vol. 15, Page 208. https://doi.org/10.3390/WEVJ15050208.

7. Bergs, Thomas, Sascha Gierlings, Thomas Auerbach, Andreas Klink, Daniel Schraknepper, and Thorsten Augspurger. 2021. "The Concept of Digital Twin and Digital Shadow in Manufacturing." *Procedia CIRP* 101: 81–4. https://doi.org/10.1016/J.PROCIR.2021.02.010.
8. Sepasgozar, Samad M. E. 2021. "Differentiating Digital Twin from Digital Shadow: Elucidating a Paradigm Shift to Expedite a Smart, Sustainable Built Environment." *Buildings* 11(4): 151: 2021, Vol. 11, Page 151. https://doi.org/10.3390/BUILDINGS11040151.
9. Becker, Fabian, Pascal Bibow, Manuela Dalibor, Aymen Gannouni, Viviane Hahn, Christian Hopmann, Matthias Jarke, et al. 2021. "A Conceptual Model for Digital Shadows in Industry and its Application." *Lecture Notes in Computer Science* 13011: 271–81: LNCS. https://doi.org/10.1007/978-3-030-89022-3_22/FIGURES/3.
10. Venkatesan, Suchitra, Krishnan Manickavasagam, Nikita Tengenkai, and Nagendran Vijayalakshmi. 2019. "Health Monitoring and Prognosis of Electric Vehicle Motor Using Intelligent-Digital Twin." *IET Electric Power Applications* 13(9): 1328–35. https://doi.org/10.1049/IET-EPA.2018.5732.
11. Mahmoud, Ibrahim, Viktor Rjabtšikov, Rassõlkin Antonet al. 2021. "Digital Twin Service Unit for AC Motor Stator Inter-turn Short Circuit Fault Detection." In *2021 28th International Workshop on Electric Drives: Improving Reliability of Electric Drives, IWED 2021 - Proceedings*. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/IWED52055.2021.9376328.
12. Hu, W., T. Wang, and F. Chu. 2023. "A Novel Ramanujan Digital Twin for Motor Periodic Fault Monitoring and Detection." *IEEE Transactions on Industrial Informatics*: 1–9. https://doi.org/10.1109/TII.2023.3248110.
13. Santos, Jhennifer F. Dos, Bendict K. Tshoombe, Lucas H. B. Santos, Ramon C. F. Araujo, Allan R. A. Manito, Wellington S. Fonseca, and Marcelo O. Silva. 2023. "Digital Twin-Based Monitoring System of Induction Motors Using IoT Sensors and Thermo-Magnetic Finite Element Analysis." *IEEE Access* 11: 1682–93. https://doi.org/10.1109/ACCESS.2022.3232063.
14. Toso, F., R. Torchio, A. Favato, P. G. Carlet, S. Bolognani, and P. Alotto. 2021. "Digital Twins as Electric Motor Soft-Sensors in the Automotive Industry." In *2021 IEEE International Workshop on Metrology for Automotive, MetroAutomotive 2021 - Proceedings*, 13–8. https://doi.org/10.1109/METROAUTOMOTIVE50197.2021.9502885.
15. Ebadpour, Mohsen, Mohammad Behdad Jamshidi, Jakub Talla, Hamed Hashemi-Dezaki, and Zdeněk Peroutka. 2023. "Digital Twin Model of Electric Drives Empowered by EKF." *Sensors* 23(4): 2006: 2023, Vol. 23, Page 2006. https://doi.org/10.3390/S23042006.
16. Amadou Adamou, Adamou, and Chakib Alaoui. 2023. "Energy Efficiency Model-Based Digital Shadow for Induction Motors: Towards the Implementation of a Digital Twin." *Engineering Science and Technology, an International Journal* 44: 101469. https://doi.org/10.1016/J.JESTCH.2023.101469.
17. Martin, Dominik, Niklas Kühl, and Gerhard Satzger. 2021. "Virtual Sensors." *Business and Information Systems Engineering* 63(3): 315–23. https://doi.org/10.1007/S12599-021-00689-W/FIGURES/4.
18. Rassõlkin, Anton, Raivo Sell, and Mairo Leier. 2018. "Development Case Study of the First Estonian Self-Driving Car, Iseauto." *Electrical, Control and Communication Engineering* 14(1): 81–8. https://doi.org/10.2478/ECCE-2018-0009.
19. Mink, F., N. Kubasiak, B. Ritter, and A. Binder. 2012. "Parametric Model and Identification of PMSM Considering the Influence of Magnetic Saturation." In *Proceedings of the International Conference on Optimisation of Electrical and Electronic Equipment, OPTIM*, 444–52. https://doi.org/10.1109/OPTIM.2012.6231768.
20. Zarate, S., G. Almandoz, G. Ugalde, J. Poza, and A. J. Escalada. 2017. "Extended DQ Model of a Permanent Magnet Synchronous Machine by Including Magnetic Saturation and Torque Ripple Effects." In *Proceedings of the 2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and Their Application to Mechatronics, ECMSM 2017*. https://doi.org/10.1109/ECMSM.2017.7945881.

Appendix 6

Publication VI

M. Ibrahim, H. A. Raja, A. Rassõlkin, T. Vaimann and A. Kallaste, "An EV-Traction Inverter Data-Driven Modelling for Digital Twin Development," 2023 23rd International Scientific Conference on Electric Power Engineering (EPE), Brno, Czech Republic, 2023, pp. 1–5, doi: 10.1109/EPE58302.2023.10149230.

2023 23rd International Scientific Conference on Electric Power Engineering (EPE) | 979-8-3503-3593-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/EPE58302.2023.10149230

An EV-Traction Inverter Data-Driven Modelling for Digital Twin Development

Mahmoud Ibrahim, Hadi Ashraf Raja, Anton Rassõlkin, Toomas Vaimann, Ants Kallaste
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
mahmoh@taltech.ee

***Abstract*— Digital Twin (DT) technology has achieved a significant breakthrough in the development of numerous applications, including electric vehicles (EVs). This technique involves the creation of a virtual version of a real object, utilizing simulation and optimization tools with real data to optimize its functioning. The traction inverter is considered the brain of the EV drive system, as it controls the power flow between the battery and the electric motor. It converts the DC voltage from the battery into AC voltage and adjusts the frequency and amplitude to control the speed and torque of the motor. This research work presents the DT modelling procedures of an EV traction inverter. The basic principles of DT are discussed, and the model-driven DT modelling technique is compared to the data-driven. The motor-drive physical model (test bench) is provided. A data-driven model of the inverter was generated using a nonlinear regression-based artificial neural network ANN. Historical data from experimental tests were used as a modeling reference. The obtained results from the data-driven model achieved high prediction accuracy.**

Keywords— Data-driven modeling, Digital twins, Model-driven development, Propulsion

I. Introduction (*Heading 1*)

Digital Twin (DT) technology is gaining attraction as it provides a comprehensive view of the physical asset and its performance. It allows for the simulation of the entire lifecycle of the physical asset, from design to operation, and provides insights into how the asset performs [1], [2]. DT technology has contributed to many applications, including EVs. This technology allows for real-time monitoring of the vehicle's performance and the ability to make changes to the vehicle's design and operation in real-time. It can be used to optimize performance and detect and diagnose any issues that may arise [3].

The DT development process is done in four phases. It starts with gathering data from various sources such as sensors, databases, and other sources. This data is then used to create a digital model of the studied system. This model is then tuned to accurately reflect the real-world system. Once the model is tuned, it can be deployed to a production environment where it can be used to monitor and analyze the system in real-time [4].

In a different context, an electric vehicle (EV) traction inverter is considered the brain of its powertrain. It is responsible for converting the (DC) power from the battery into (AC) power, which is then used to power the electric motor. The inverter also controls the torque and speed of the motor, allowing the driver to control the acceleration and deceleration of the vehicle. It also helps to protect the battery from overcharging and over-discharging, ensuring the battery is not damaged by excessive current [5].

DTs are playing an effective role in the development of traction inverters. By creating a virtual model of the inverter, engineers can simulate and analyze the performance of the inverter in various conditions, allowing them to optimize its design and performance. The literature surveyed reveals that there is a great interest in developing DT models for EV traction inverters and their components. They were mainly used for health monitoring, fault diagnosis, performance optimization, and lifetime estimation of semiconductors. Wunderlich and Santi [6] developed a dynamic neural network-based DT model of an EV inverter for condition monitoring. A particle swarm optimization algorithm for the parameter identification of an IGBT-based three-phase DC/AC inverter for condition monitoring was proposed in [8]. Subsequently, Liu et al. [7] created and tested experimentally a DT model for an automotive Silicon carbide (SiC) based traction drive system.

This paper proposes the DT modeling procedures of an EV traction inverter. It's organized as follows. Section I addresses an introduction to DT technology and the main research topic. Section II. A illustrates the traction inverter physical model tended to be modeled, and Section II.B describes the inverter modeling procedures based on a data-driven approach. Section III highlights the research results. Section IV addresses the research conclusions and future direction.

II. Modelling Procedures

DT comprises three main components: a physical model, a virtual model, and a communication link between the two. The physical model represents the physical object or system, such as a test bench. The virtual model is a digital representation of the physical object or system. The communication link between the two is a connection between the physical and virtual models, allowing them to exchange data. This data can be used to monitor the physical object or system and make predictions about its future behavior.

A. Traction Inverter Physical Model

The physical inverter model was taken from ISEAUTO [8] (55-kW heavy-duty traction inverter unit) with a hybrid field-oriented control (FOC) and direct torque control (DTC) algorithm). It drives a Y4F1 Interior permanent magnet synchronous motor IPMSM. A visual interface PC unit controls the EV inverter, while a battery emulator unit combined with a regenerative AC to DC converter is set to simulate the behavior of real EV batteries. A four-gear double-stage type transmission unit (F1E1A) is connected to the motor shaft and coupled with a differential unit. Two 7.5 kW loading induction motors (IM) are mounted on both sides of

The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of an autonomous electric vehicle".

the differential to simulate the motion of the wheels. An ABB frequency converter controls the loading motors, as shown in Fig. 1 shows the testbench setup.

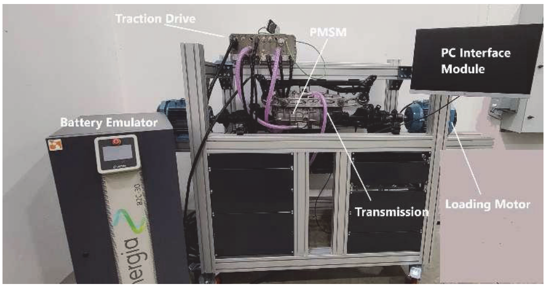


Fig. 1. EV drive system testbench

A modular power analyzer and data acquisition system (DAS) was used to measure the inverter inputs of DC bus voltage and current and outputs of 3-phase voltage and currents. The speed feedback is measured from the motor side using a resolver unit mounted on the motor shaft. The loading torque is estimated through a virtual torque sensor.

B. Modelling Techniques

DT modeling process can be done in two ways according to the modelled system complexity and the DT function. Data-driven and Model-driven modeling techniques are the two approaches used for DT modeling. Data-driven modeling techniques involve various methods, such as regression analysis, decision trees, and neural networks. At the same time, model-driven modeling techniques are based on the idea that the system can be represented as a set of models or mathematical equations that can be used to describe the system's behavior. Both approaches have their advantages and disadvantages. Data-driven modeling is more accurate and can be used to model complex systems, but it is also more time-consuming and expensive. Model-driven modeling is faster and cheaper, but it is preferred only to be used to model simpler systems [9].

In a connected context, modeling an EV traction inverter is a complex process that requires a deep understanding of the underlying physics and electrical engineering principles. It involves the analysis of the electrical components of the inverter, such as the power switches, the gate drivers, the control algorithms, the power stage, and the cooling system.

Despite the presence of the actual model of the inverter in the drive system testbench, predicting what is inside it or the control algorithm on which the process is based is very complex.

C. Data Collection

The first step is to identify the modeled system's inputs and outputs. The input of a traction inverter is typically a DC voltage from the battery, which is then converted to a three-phase AC voltage. The output of the traction inverter is a three-phase AC voltage, which is then fed to the electric motor. The inverter also has a control input, which is used to control the speed and torque of the motor. The following graph is an illustrative figure.

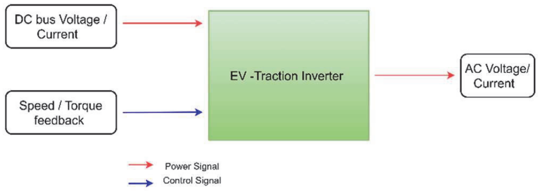


Fig. 2. EV traction inverter inputs and outputs.

Data collection for a data-driven model of a traction inverter involves gathering information about the inverter's performance, such as its power output, efficiency, and temperature. This data can be collected through a variety of methods, including experimental testing.

Different loading / no-load (% EV-motor torque) tests with variable speeds were done on the test bench to collect suitable data for the modeling process. Table I illustrates different test conditions.

TABLE I. TEST CASE SET 1

Case	Load condition	Speed (rpm)
A	No load (5%)	200-400-600-800-1000
B	Light load (20%)	200-400-600-800-1000
C	Medium load (40%)	200-400-600-800-1000
D	Heavy load (80%)	200-400-600-800-1000

DAS was used to measure and pre-record the input and output data to and from the traction inverter during the tests. It was also used to collect the actual speed feedback data from the resolver unit. The virtual torque sensor model extracted the real-time estimated feedback torque data. The collected data of each test was stored as matrices in MATLAB file form.

D. Traction Inverter Data-Driven Modelling

The data-driven model of the inverter was built using a deep artificial neural network (ANN) based on a nonlinear regression approach using the MATLAB platform. The nonlinear regression learning approach is according to the following equation:

$$Ki = h\left[xi(1), xi(2), \dots, xi(m)\, ;\, \sigma 1, \sigma 2, \dots, \sigma p\right] + E \quad (1)$$

where, K – number of responses, h - the function, x - inputs, σ - parameters being estimated, and E - error term.

The ANN can recognize complex data patterns by using multiple layers of neurons to process and analyze the input data. Each layer of neurons is responsible for extracting a different set of features from the data, and the layers are connected in a way that allows them to pass information from one layer to the next.

The ANN was trained using a set of input-output collected data. The data set included the input voltage and current, the inverter's speed and torque feedback, output voltage, and output current. The ANN was trained using a backpropagation algorithm with a learning rate of 0.01 and a momentum of 0.9. Fig. 3 shows the architecture of the used ANN.

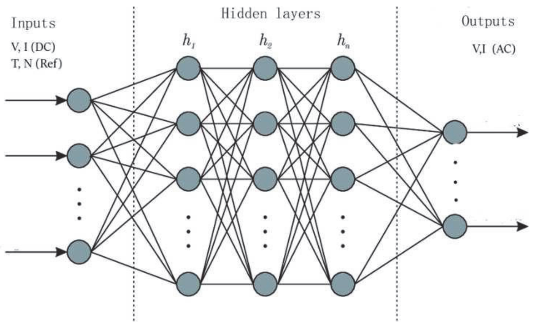


Fig. 3. Deep ANN architecture used for data-driven modeling.

Input data were clustered into four inputs (A, B, C, D) as done in tests. Each input comprises five sub-inputs representing different testing cases. Every sub-input includes four variables of DC voltage and current, speed, and torque. Output data were clustered into four responses according to testing cases and two AC voltage and current sub-responses. The two sub-responses include six variables of three phases voltage and currents.

Three types of ANNs (wide ANN, Billiard ANN, and, Optimized ANN with a grid search optimizer) were used for model generation to provide more choices and increase the model's overall accuracy.

The wide ANN was used to generate a model with a large number of parameters, allowing for more flexibility in the model. The billiard ANN was used to generate a model with fewer parameters but with more emphasis on the model's accuracy. Finally, the optimized ANN with a grid search optimizer was used to generate a model with the best combination of parameters and accuracy. Figs 4-6. Shows the model prediction over true values results of the three ANN types.

The results of the model generation process showed that the Billiard ANN had the highest accuracy over the Wide and Optimized ANNs however, fig 4 showed higher coverage of predicted points over true ones with the Optimized ANN. Although the Root Mean Squared Error (RMSE) for the Billiard ANN is lower than that of the Optimized ANN, the focus on accuracy in the Billiard ANN led to the omission of some parameters. This does not provide the optimal solution, as the predicted output is limited, and if validation is conducted with all parameters included, accuracy is reduced.

Finally, the selected model was extracted as a compact MATLAB function able to handle input data sets and make a prediction of the outputs.

III. Results and Discussion

The generated model was verified through two verification steps. The first verification stage involves running the generated ANN model with the same input data used to generate the model. This allows the model to be tested against known results. The second verification stage involves running the model with new input data and comparing the results to the expected output. This allows the model to be tested against unknown results. This process helps to ensure that the model is accurate and reliable.

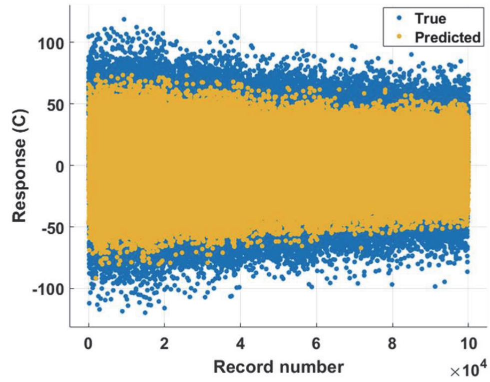


Fig. 4. Model 1, optimized ANN. (RMSE 14.06, Accuracy 85.94%)

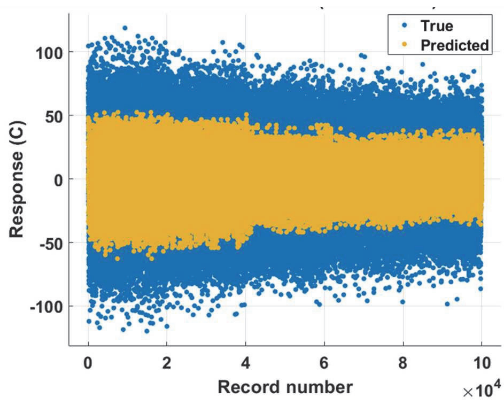


Fig. 5. Model 1, billiard ANN. (RMSE 13.27, 86.73%).

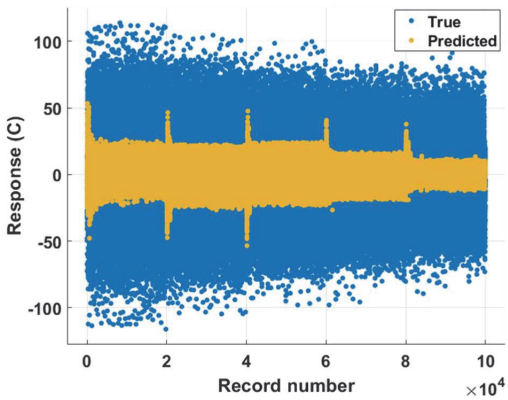
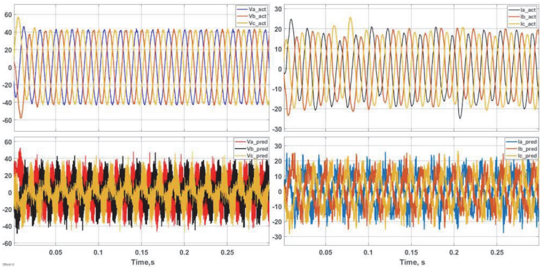


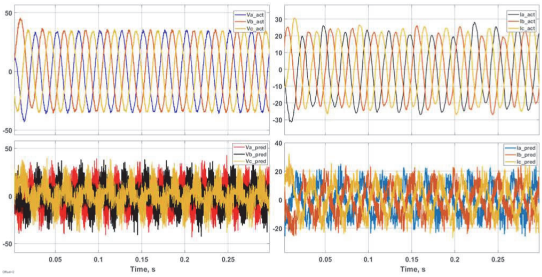
Fig. 6. Model 1, wide ANN. (RMSE 14.81, Accuracy 85.19%)

A. 1st Verification Stage

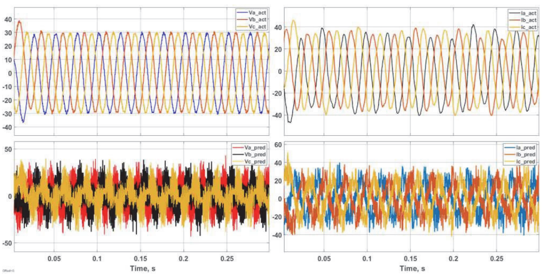
The input data from test cases, mentioned in Table I, was used to check the model accuracy in predicting the outputs. The predicted output data were compared to the actual output data of the test case. Figs. 7(a-d) compare actual and predicted outputs of different cases.



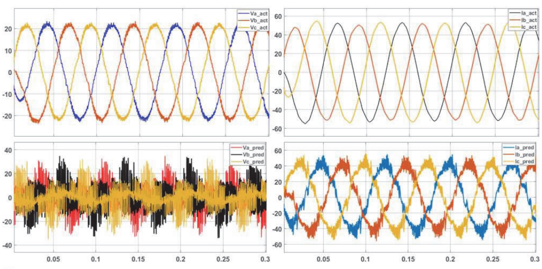
(a)



(b)



(c)



(d)

Fig.7. Comparison between actual and ANN model predicted output AC voltage and current for training cases: a – Case A at 1000 rpm; b – Case B at 800 rpm; c – Case C at 600 rpm; d – Case D at 400 rpm.

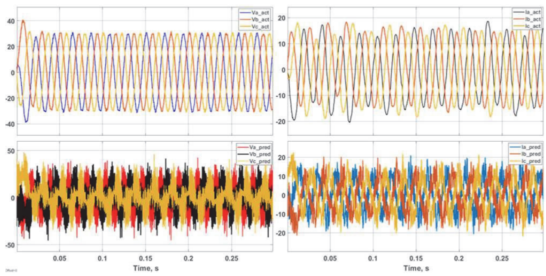
It is noticeable from Figure 6 that the predicted values from the ANN model are noisy compared to the actual values, but the RMS value for both is nearly the same (error less than 10% in most cases), which can be explained by the fact that the RMS value is a measure of the average magnitude of the errors in a set of predictions, rather than the amount of noise in the predictions.

B. 2nd Verification Stage

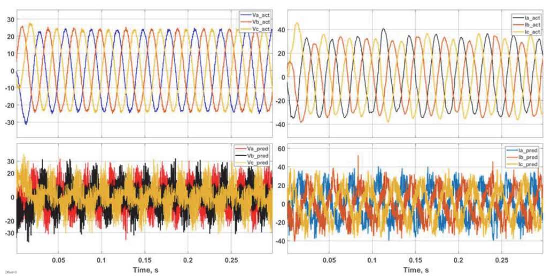
The second verification stage involves testing the ANN model on a new data set. This input data is taken from new test cases from the one used to train the model. This stage aims to evaluate the model's ability to generalize and accurately predict responses to unseen data. The model's performance is then compared to the actual data to assess the model's accuracy and reliability. Table II illustrates new test conditions, and Fig. 8(a-c) compares actual and predicted outputs from new cases.

TABLE II. TEST CASE SET 2

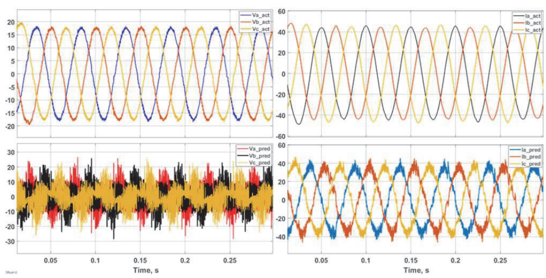
Case	Load condition	Speed (rpm)
E	No load (5%)	300-500-700-900
F	Medium load (40%)	300-500-700-900
G	Heavy load (80%)	300-500-700-900



(a)



(b)



(c)

Fig.8. Comparison between actual and ANN model predicted output AC voltage and current of new cases: a – Case E at 700 rpm; b – Case F at 500 rpm; c – Case G at 1000 rpm.

From Fig. 8, although the outputs are still noisy, the RMS value of both actual and predicted values are nearly the same (error less than 10%), which verifies the model's ability to predict unseen data with different input parameters.

Additionally, it proves the ability of the ANN model to capture the data pattern and generalize the relations between variables.

IV. Conclusions

The paper highlighted the modeling process as the first stage of DT development. DTs can be modeled in two ways, as follows:

1. Data-driven modeling - this approach uses data from sensors and other sources to create a model of the system. This model is then used to predict the behavior of the system.
2. Model-driven modeling - this approach uses existing system models to create a DT. This model is then used to simulate the behavior of the system.

However, it should be mentioned that the hybrid approach of DT development might combine the benefits of both modeling approaches.

The main research topic was the modeling procedures of an EV traction inverter system based on a data-driven technique for DT development. A nonlinear-based ANN was used to generate the inverter model using data sets of the inputs and outputs from experimental tests. The trained ANN inverter model was verified through two verification steps depending on the previously used training data and new unseen data. The verification results showed that the generated ANN model of the traction inverter could capture the data pattern and generalize the relationship between the inverter inputs and outputs with an overall accuracy of 85.49%.

It's recommended for future works to add more training data sets of different cases to cover the whole operation of the inverter, which would enhance the model performance and increase its accuracy.

References

[1] M. Ibrahim, A. Rassõlkin, T. Vaimann, and A. Kallaste, "Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System," Sustain. 2022, Vol. 14, Page 601, vol. 14, no. 2, p. 601, Jan. 2022, doi: 10.3390/SU14020601.

[2] O. San, "The digital twin revolution," Nat. Comput. Sci. 2021 15, vol. 1, no. 5, pp. 307–308, May 2021, doi: 10.1038/s43588-021-00077-0.

[3] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital Twin: Enabling Technologies, Challenges and Open Research," IEEE Access, vol. 8, pp. 108952–108971, 2020, doi: 10.1109/ACCESS.2020.2998358.

[4] R. He, G. Chen, C. Dong, S. Sun, and X. Shen, "Data-driven digital twin technology for optimized control in process systems," ISA Trans., vol. 95, pp. 221–234, Dec. 2019, doi: 10.1016/J.ISATRA.2019.05.011.

[5] X. Liu, M. Hofmann, F. Streit, and M. Maerz, "Digital Twin for Intelligent and SiC-based Drive Systems," 2021 11th Int. Electr. Drives Prod. Conf. EDPC 2021 - Proc., 2021, doi: 10.1109/EDPC53547.2021.9684201.

[6] A. Wunderlich and E. Santi, "Digital twin models of power electronic converters using dynamic neural networks," Conf. Proc. - IEEE Appl. Power Electron. Conf. Expo. - APEC, pp. 2369–2376, Jun. 2021, doi: 10.1109/APEC42165.2021.9487201.

[7] H. Shi, L. Xiao, Q. Wu, and W. Wang, "Digital Twin Approach for IGBT Parameters Identification of a Three-Phase DC-AC Inverter," 2022 IEEE Transp. Electrif. Conf. Expo, Asia-Pacific (ITEC Asia-Pacific), pp. 1–4, Oct. 2022, doi: 10.1109/ITECASIA-PACIFIC56316.2022.9942105.

[8] Rassõlkin, Anton; Sell, Raivo; Leier, Mairo; ‚Development case study of first Estonian Self-driving car ISEAUTO,"The Scientific Journal of Riga Technical University-Electrical, Control and Communication Engineering",14,1,81-88,2018.

[9] M. Ibrahim, V. Rjabtšikov, and R. Gilbert, "Overview of Digital Twin Platforms for EV Applications," Sensors 2023, Vol. 23, Page 1414, vol. 23, no. 3, p. 1414, Jan. 2023, doi: 10.3390/S23031414.

Appendix 7

Publication VII

M. Ibrahim, A. Rassõlkin and V. Rjabtšikov, “Reverse Engineering-Based Modeling of an EV Motor Drive for Digital Twin Development,” 2024 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference (ESARS-ITEC), Naples, Italy, 2024, pp. 1–5, doi: 10.1109/ESARS-ITEC60450.2024.10819839.

Reverse Engineering-Based Modeling of an EV Motor Drive for Digital Twin Development

Mahmoud Ibrahim
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
mahmoud.mohamed@taltech.ee

Anton Rassõlkin
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
anton.rassolkin@taltech.ee

Viktor Rjabtšikov
Department of Electrical Power Engineering and Mechatronics
Tallinn University of Technology
Tallinn, Estonia
viktor.rjabtsikov@taltech.ee

2024 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference (ESARS-ITEC) | 979-8-3503-7390-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/ESARS-ITEC60450.2024.10819839

***Abstract*— Digital twins (DTs) play crucial roles in various applications, with their significance extending to critical domains like electric vehicles (EVs). Essentially, a DT is a virtual representation or digital counterpart of a physical object, system, or process, crafted through sensor technology, data collection, modeling, and simulation. Modeling a real system with uncertain parameters is a challenging task. This paper presents a comprehensive methodology for reverse engineering the drive system of an EV Permanent Magnet Synchronous Motor (PMSM) with an uncertain control strategy parameter. The motor was modeled based on d-q mathematical representation. Extensive tests on the EV motor drive under diverse conditions and scenarios were conducted, with performance data acquired using a precision data acquisition system (DAS). The collected data were preprocessed and refined for the modeling process. In response to feedback on input mechanisms, a data-driven model employing a Long Short-Term Memory (LSTM) network was developed to emulate the drive system's behavior. The LSTM-based drive system model underwent rigorous training with the actual data collected from the physical system. The overall accuracy of the LSTM model in replicating the motor drive system was 96.7%, affirming its high performance and reliability in mimicking the intricate drive system. The LSTM model was then combined with the motor simulation model, creating a hybrid-driven model. The hybrid model's success in closely aligning with the actual motor drive outputs underscores its comprehensive ability to effectively reflect both the nuanced dynamics and the operational intricacies of the EV propulsion system.**

Keywords—Digital twin, electric vehicle, drive system

I. Introduction

In the dynamic advancing field of EVs, the adoption of DT technology marks a significant shift in engineering practices. DTs were originally proposed by Dr. Michael Grieves at the University of Michigan in 2002, DT technology has since proliferated across various industries, with its integration into the automotive sector being particularly impactful [1]. As global efforts to electrify transportation intensify, DTs have become instrumental in improving the design, development, performance optimization, and maintenance of EVs [2]. At its core, the DT concept involves creating a virtual model of a physical object, system, or process that is continuously updated with real-time data from its physical counterpart, making the virtual model a dynamic entity[3]. The implementation of a DT for a physical system brings numerous advantages. It enhances understanding of the system's behavior, facilitates predictive analytics, and supports real-time monitoring. These capabilities translate into cost reductions, more streamlined design processes, and the ability to operate systems remotely [4].

The modeling phase is the most crucial step in building a DT of a system. Accurately modeling a preexisting system is particularly challenging when some of its components experience a degree of uncertainty. This uncertainty can stem from incomplete knowledge about the system's internal workings or from unknown parameters, which complicates the process of creating an accurate and reliable model. In the case of motor drives, this opacity regarding their inner mechanisms poses a significant obstacle. Without a clear understanding of all relevant parameters, it becomes difficult to replicate the system's behavior with high fidelity. As highlighted in [5] the lack of transparency poses a substantial obstacle in the pursuit of system optimization. Overcoming these challenges is essential for developing a functional DT that can provide valuable insights and optimize system performance.

In such a context, the concept of reverse engineering becomes highly applicable and valuable. Reverse engineering involves thoroughly analyzing the structure, functions, and operations of a physical system to determine its properties [6]. It enables manufacturers to gain insights into the original design of a part, facilitating replication, modification, or enhancement. It has recently gained high attention as a critical practice in research, driving innovation and enabling a deeper understanding of complex systems [7].

Reverse engineering techniques find valuable applications in EV research, helping to dissect and comprehend the inner workings of EV components and systems. This process drives advancements, innovations, and optimizations in the industry. Li et al. [8] utilized the application of reverse engineering to decode the design of an EV front panel. Their work involved meticulous analysis of the panel's inputs and outputs, leading to the development of an emulated control strategy. Millo et al. [9] introduced a new method to reverse engineer a P2 Plug-in Hybrid EV's controller, using various data sources such as an on-board diagnostic system, control area network, and additional sensors on the vehicle high voltage network. Their results demonstrate a 1-2% error in predicting CO2 emissions and successfully matching state of charge (SOC) profiles. Panchal et al. [10] explored reverse engineering in the design of water-cooled battery systems for EVs, using a high Reynolds number turbulent model for micro-channel cold plates, aiding in the development of efficient battery cooling systems.

The research has been supported by the Estonian Research Council under grant PSG453 "Digital twin for propulsion drive of an autonomous electric vehicle".

This research addresses a crucial aspect of the DT modeling of an EV propulsion system by introducing a methodology to reverse engineer the motor drive system. The primary issue addressed is the presence of partially unknown parameters within the drive system. The study adopts a comprehensive approach, rigorously testing the EV motor drive system under various conditions and collecting performance data using a precision data acquisition system (DAS). This departure from established paradigms fills a critical void in DT development and offers a novel perspective on modeling techniques, particularly when there is uncertainty in the physical model's parameters.

The paper is structured as follows: Section I offers a systematic review of recent advancements in reverse engineering methodologies applied to EV applications. In Section II, a comprehensive case study of the motor drive system is presented. Section III delves into procedures of the reverse engineering process of the drive system. Section IV highlights the resulting hybrid model of the EV motor drive system. The conclusions drawn from the research are summarized in Section V.

II. EV-PMSM Drive System

A. EV-PMSM Drive Physical Model

The EV motor's physical model employed in this research has its origins in the innovative ISEAUTO project, an Estonian Autonomous EV designed with a primary focus on serving as a last-mile transportation solution within the confines of the Tallinn University of Technology (TalTech) campus. ISEAUTO propulsion system was built based on a commercial YF4 PMSM coupled with a four-stage transmission unit. For a comprehensive understanding of this motor system's characteristics, Table 1 provides the parameters of the ISEAUTO motor drive system.

TABLE I. UNDER STUDY PMSM PARAMETERS

Parameter	Description	Value	Unit
p	Number of pole pairs	4	-
Rs	Stator resistance	0.10087	Ω
Lq	q-axis inductance	2.4·10-4	H
Ld	d-axis inductance	4.4·10-4	H
λpm	Permanent magnet linkage flux	0.087	Wb
Nr	Rated speed	3000	rpm
Tmax	Max torque	180	N·m

The motor is driven by a traction inverter featuring insulated-gate bipolar transistors (IGBT) and combined vector control with flux weakening strategies. The control strategy is characterized by a degree of uncertainty concerning its specific parameters and structural elements. This uncertainty constitutes a central research challenge, as the primary objective of the investigation is to establish a precise model of the motor's drive system, which serves as a foundational step for the subsequent creation of the EV powertrain DT.

B. PMSM Mathematical Model

Since the motor parameters are known, conventional model-based techniques can be used for its modeling, relying on standard mathematical representations. The d-q equivalent circuit, a widely used technique in electric motor analysis, transforms the three-phase PMSM into two orthogonal components: the direct axis (d), which represents the magnetic field generated by the permanent magnet, and the quadrature axis (q), which represents the flux produced by the stator current. [11].

The stator voltage equations can be written in synchronous *d–q* reference frame as follows:

$$v_d = ri_d + \frac{d}{dt}\lambda_d - \omega_e\lambda_q, \quad (1)$$

$$v_q = ri_q + \frac{d}{dt}\lambda_q + \omega_e\lambda_d, \quad (2)$$

Where the flux linkage equation can be given below.

$$\lambda_d = \lambda_{pm} + l_d i_d, \quad (3)$$

$$\lambda_q = l_q i_q, \quad (4)$$

$$T_e = \frac{3}{2} \cdot p \cdot i_q(\lambda_d - \lambda_q). \quad (5)$$

Mechanical output equation

$$J \cdot d\omega r/dt = T_e - T_l - B \cdot \omega_r, \quad (6)$$

where,

v_d, v_q are stator voltage components in d-q rotating frame;

i_d, i_q are stator voltage components in d-q rotating frame;

λ_d, λ_q are stator flux components in d-q rotating frame;

r is stator resistance.

ω_e, ω_r is rotor electrical speed, rotor mechanical speed.

λ_{pm} is permanent magnet linkage flux.

l_d, l_q are stator inductance components in d-q rotating frame;

T_e is motor electrical torque.

J is motor inertia.

B is the friction coefficient.

III. EV-Drive System Reverse Engineering Modelling

The process of reverse engineering begins with a delineation of its constituent blocks, each representing a distinct functional unit within the overall mechanism. This initial step involves identifying and understanding the roles, inputs, and outputs of these blocks, thereby laying the groundwork for a comprehensive analysis of the system's architecture. Fig. 1 shows the architecture of the EV-drive system.

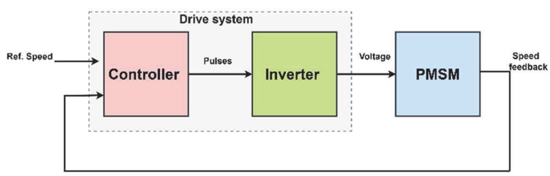


Fig.1. EV-drive system block diagram.

The main inputs to the model are the reference and feedback speed signals, which are the directives provided to the real drive system to achieve desired operational states. Current and voltage, are considered the immediate control actions taken by the system. The next critical step involves experimentally testing the drive system to gather extensive data on its behavior and dynamics.

A. Experimental tests

Comprehensive tests were conducted on the EV motor drive under both controlled and diverse conditions, enabling a comprehensive assessment of its performance. During these tests, the EV motor drive was equipped with voltage, current, reference speed ramp signal, and motor speed sensor directly connected to the DAS system. This setup facilitated real-time data gathering, laying the groundwork for detailed analysis and subsequent reverse engineering processes. The vehicle was driven under different conditions in both forward and backward directions, accounting for the varied operational conditions. Table II addresses the test conditions.

TABLE II. EV TEST CONDITIONS

Test	Motor Speed (P.U.)	Condition
a	0.3	Forward
b	0.59	Forward
c	0.53	Backward
d	0.72	Backward
e	0.88	Forward
f	0.91	Backward
g	0.97	Forward and Backward

The dataset obtained from the series of tests boasts collection methods, featuring a high sampling frequency of 10 KHz, ensuring a remarkable level of accuracy. Tests a,b,c,e, and g were targeted for the model training while cases d and g were reserved as unseen data for model validation and testing.

B. Data Preprocessing

After collecting data from the experimental tests, the subsequent step is to preprocess this data through refining, cleaning, and filtering. This stage is pivotal as it transforms raw experimental data into a coherent and usable format, removing any anomalies or irrelevant information that could skew the analysis. The refinement process includes filtering out noise, correcting errors, and handling missing values, ensuring the data accurately reflects the system's actual performance. Following this, an advanced regression correlation technique is employed. This thorough data processing step is essential for uncovering the underlying dynamics of the drive system, setting a solid foundation for developing accurate models.

C. LSTM Architecture

The Long short-term memory (LSTM) network consists of multiple layers, including an input layer, eight hidden LSTM layers, and an output layer. The input layer is designed to receive time-series data, specifically the motor's reference (desired) speed and the actual speed feedback, which is intentionally delayed to simulate real-world operational lag and test the network's predictive capabilities. The hidden layers, composed of LSTM units, are where the network learns to recognize patterns in the input data, accounting for long-term dependencies that traditional networks might overlook. Each LSTM unit includes a cell state and three gates (input, output, and forget gates), which regulate the flow of information into and out of the cell, allowing the network to make informed predictions based on both recent and historical data. The architecture unfolds into four hidden layers. The uniform size of these layers ensures a consistent capacity for feature extraction and transformation, facilitating the network's ability to handle the sophisticated dynamics of the drive system. The output layer is configured to produce the predicted motor voltage necessary to achieve the desired speed, effectively closing the loop in the control strategy.

The chosen activation functions within the LSTM units—typically tanh for modulating the cell states and sigmoid for the gates—remain integral to managing the flow of information through the network, ensuring that the LSTM can effectively learn from both recent and long-term dependencies. To combat overfitting, dropout is applied within the hidden layers.

Employing Adam optimizer alongside the mean squared error (MSE) loss function, the network is trained to encapsulate the nuanced relationship between the input features and the target outputs. Fig. 2. Illustrative diagram of the LSTM.

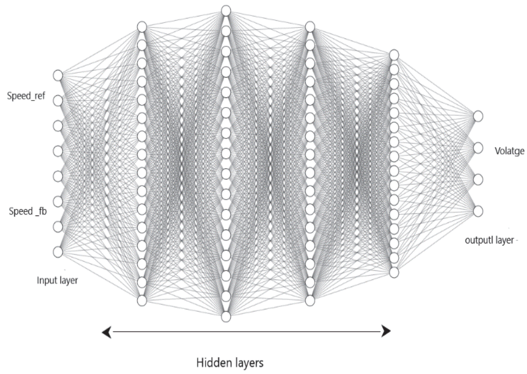


Fig.2. LSTM archtecture.

D. LSTM Training

The model was created using MATLAB machine learning toolbox, the dataset was partitioned into three subsets: training (65%), validation (20%), and testing (15%). The training set, constituting most of the data, was employed to facilitate the learning process of the LSTM network by adjusting the model's parameters through the mechanism of backpropagation based on the discrepancies between the predicted and actual values. The validation set was instrumental in the fine-tuning of hyperparameters and in making informed decisions regarding adjustments to the model architecture, serving as a tool to mitigate overfitting by providing feedback on the model's performance with unseen data during the training phase. Finally, the testing set was utilized to assess the model’s performance post-training, offering a crucial evaluation of its generalization capabilities on new, unseen data. As depicted in Fig. 3, the model demonstrated a high level of agreement between the outputs and the targets over time, alongside a notable reduction in prediction error, underscoring the LSTM’s efficacy in capturing and predicting the underlying patterns in the dataset.

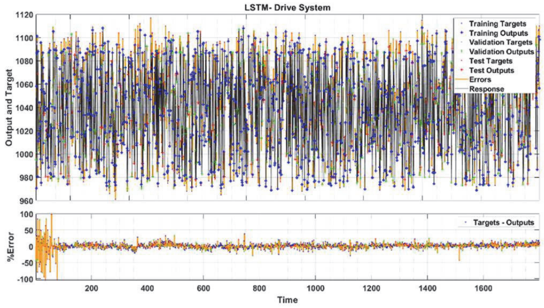


Fig.3. LSTM performance

The RMSE value stabilized at 0.033, which underscores the high precision of the LSTM model's effectiveness in capturing the essential dynamics and trends of the dataset.

IV. Results and Discussion

The training process of the LSTM network reached a remarkable conclusion, achieving an accuracy rate of 96.7%. This level of accuracy highlights the model's capability to accurately capture the intricate dynamics of the drive system. It reflects a profound comprehension of the complex interrelations between the input variables and the system's output requirements. The effectiveness of the LSTM model is further demonstrated by its integration into a comprehensive simulation model of the motor drive system, as depicted in Fig 4.

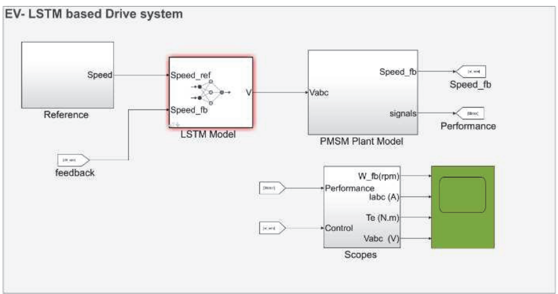


Fig.4. EV LSTM-based drive system model.

The Simulink model depicted in the above Fig. 4 is intricately designed to simulate the performance of the motor drive system using the developed LSTM network. The first block in the model represents the reference speed signal, which serves as the primary input to the system. The LSTM block processes the inputs and outputs a voltage signal tailored to achieve the desired motor performance.

Following the LSTM block is the motor plant model. This component is developed based on the mathematical expressions of the PMSM, which were detailed previously in section II. The motor plant model effectively simulates the motor's physical response to the voltage inputs derived from the LSTM. The output from the motor plant model is the actual speed signal of the motor, which is crucial for feedback. This speed signal is utilized as a delayed input to the LSTM, forming a feedback loop that allows the LSTM to continuously update and refine its predictions based on the latest motor performance data. This feedback mechanism is vital for the LSTM to adapt and optimize its output.

To rigorously evaluate the performance of the developed LSTM-based simulation model, a comprehensive assessment was conducted using unseen data from test cases "d" and "f" from section III. This methodology was essential to validate the model's predictive accuracy and its generalizability to new scenarios, providing a robust test of the simulation model's effectiveness in real-world settings. The results of this evaluation are visually represented in Figs 7 and 8, which illustrate the comparison between the steady state predicted voltage outputs from the LSTM model and the actual voltage signals measured in the motor drive system for both test cases.

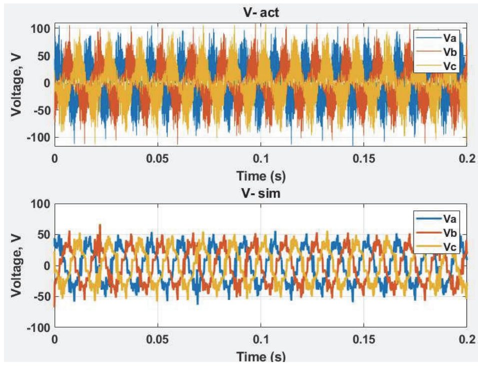


Fig.5. Actual vs predicted drive system output voltage - case d.

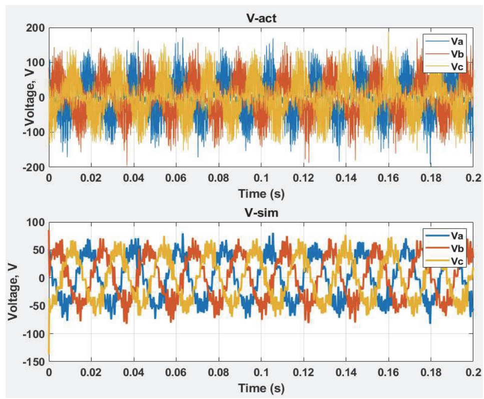


Fig.6. Actual vs predicted drive system output voltage - case F.

From the detailed analysis of Figs 5 and 6, it is evident that there is a high degree of correspondence between the predicted and actual RMS voltage signals of the drive system for both test cases d and f. This close alignment underscores the LSTM model's success in capturing the essential dynamics of the motor drive system, enabling it to precisely predict the voltage requirements across different operational scenarios. However, it is important to note that despite the overall high fidelity of the model's predictions, there are observable harmonics in the actual voltage signals, as well as some noise artifacts. These discrepancies could be attributed to various factors inherent in real-world systems, such as electromagnetic interference, fluctuating load conditions, or imperfections in the physical components of the drive system.

V. Conclusions

In this study, a comprehensive reverse engineering technique was employed to model an EV motor drive system incorporating a degree of uncertainty that marks a crucial step in the development of the DT the entire EV propulsion system. This reverse engineering approach was grounded in a data-driven methodology using an LSTM network, specifically tailored to capture the dynamic behavior of the motor drive system under various operating conditions. The LSTM network was trained on a substantial dataset derived from real-world EV drive system operations. After extensive training and validation phases, the LSTM model achieved a high accuracy of 96.7%, demonstrating its proficiency in accurately predicting the system's performance. To enhance the sophistication and utility of the reverse-engineered model, a hybrid approach was implemented. This involved integrating the LSTM-based data-driven model with a physics-based motor model. Hybrid models, which combine the strengths of data-driven and physics-based approaches, offer significant advantages in DT development. They provide faster response times and are more suitable for real-time applications compared to high-fidelity models, which, despite their accuracy, often require longer computational times that are impractical for real-time analysis. This capability is particularly critical in the context of DTs, where real-time monitoring, simulation, and decision-making are essential. The hybrid model thus not only maintains a high level of accuracy but also ensures the responsiveness required for effective DT applications. Future work will focus on expanding the LSTM model's robustness and accuracy by integrating more diverse scenarios and datasets, including transient data crucial for capturing rapid dynamic changes in the motor drive system.

References

[1] O. San, “The digital twin revolution,” Nat. Comput. Sci. 2021 15, vol. 1, no. 5, pp. 307–308, May 2021, doi: 10.1038/s43588-021-00077-0.

[2] M. Ibrahim, A. Rassõlkin, T. Vaimann, and A. Kallaste, “Overview on Digital Twin for Autonomous Electrical Vehicles Propulsion Drive System,” Sustain. 2022, Vol. 14, Page 601, vol. 14, no. 2, p. 601, Jan. 2022, doi: 10.3390/SU14020601.

[3] M. G. Juarez, V. J. Botti, and A. S. Giret, “Digital twins: Review and challenges,” J. Comput. Inf. Sci. Eng., vol. 21, no. 3, Jun. 2021, doi: 10.1115/1.4050244/1100428.

[4] M. Grieves and J. Vickers, “Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems,” Transdiscipl. Perspect. Complex Syst. New Find. Approaches, pp. 85–113, Jan. 2016, doi: 10.1007/978-3-319-38756-7_4/COVER.

[5] G. Bianchin, M. Vaquero, J. Cortes, and E. Dall’Anese, “Online Stochastic Optimization for Unknown Linear Systems: Data-Driven Synthesis and Controller Analysis,” Aug. 2021, Accessed: Sep. 21, 2023. [Online]. Available: https://arxiv.org/abs/2108.13040v1.

[6] M. G. Rekoff, “On Reverse Engineering,” IEEE Trans. Syst. Man Cybern., vol. SMC-15, no. 2, pp. 244–252, 1985, doi: 10.1109/TSMC.1985.6313354.

[7] E. J. Chikofsky and J. H. Cross, “Reverse Engineering and Design Recovery: A Taxonomy,” IEEE Softw., vol. 7, no. 1, pp. 13–17, 1990, doi: 10.1109/52.43044.

[8] S. Freiberger, M. Albrecht, and J. Käufl, “Reverse Engineering Technologies for Remanufacturing of Automotive Systems Communicating via CAN Bus,” J. Remanufacturing, vol. 1, no. 1, pp. 1–14, Dec. 2011, doi: 10.1186/2210-4690-1-6/FIGURES/9.

[9] G. Dipierro, E. Galvagno, G. Mari, F. Millo, M. Velardocchia, and A. Perazzo, “A Reverse-Engineering Method for Powertrain Parameters Characterization Applied to a P2 Plug-In Hybrid Electric Vehicle with Automatic Transmission,” SAE Int. J. Adv. Curr. Pract. Mobil., vol. 3, no. 1, pp. 715–730, Jun. 2020, doi: 10.4271/2020-37-0021.

[10] S. Panchal, K. Gudlanarva, M. K. Tran, R. Fraser, and M. Fowler, “High Reynold’s Number Turbulent Model for Micro-Channel Cold Plate Using Reverse Engineering Approach for Water-Cooled Battery in Electric Vehicles,” Energies 2020, Vol. 13, Page 1638, vol. 13, no. 7, p. 1638, Apr. 2020, doi: 10.3390/EN13071638.

[11] M. Ibrahim, V. Rjabtsikov, S. Jegorov, A. Rassolkin, T. Vaimann, and A. Kallaste, “Conceptual Modelling of an EV-Permanent Magnet Synchronous Motor electric,” 2022 IEEE 20th Int. Power Electron. Motion Control Conf., pp. 156–160, Sep. 2022, doi: 10.1109/PEMC51159.2022.9962943.

Curriculum vitae

Personal data

Name:	Mahmoud Ibrahim
Date of birth:	20.11.1993
Place of birth:	Zagazig, Egypt
Citizenship:	Egyptian

Contact data

E-mail:	mahmoud.mohamed@taltech.ee

Education

2021–2025	Tallinn University of Technology, Energy Conversion and Control Systems, PhD studies
2018–2021	Zagazig University, Electrical Engineering, MSc
2011–2016	Zagazig University, Electrical Engineering, BSc

Language competence

English	Fluent
Estonian	Intermediate
German	Basic
Arabic	Native

Professional employment

01.09.2022–...	Tallinn University of Technology, School of Engineering, Department of Electrical Power Engineering and Mechatronics, Junior Researcher (1,00)
1.01.2021–30.09.2021	Egyptian Russian University, School of Engineering, Department of Telecommunication and Electronics, Assistant Lecturer (1,00)
01.10.2018–31.12.2020	Egyptian Russian University, School of Engineering, Department of Telecommunication and Electronics, Teaching Assistant (1,00)

Elulookirjeldus

Isikuandmed

Nimi:	Mahmoud Ibrahim
Sünniaeg:	20.11.1993
Sünnikoht:	Zagazig, Egiptus
Kodakondsus:	Egiptuse

Kontaktandmed

E-post:	mahmoud.mohamed@taltech.ee

Hariduskäik

2021–2025	Tallinna Tehnikaülikool, Elektroenergeetika ja mehhatroonika, PhD
2018–2021	Zagazigi Ülikool, Elektrotehnika, MSc
2011–2016	Zagazigi Ülikool, Elektrotehnika, BSc

Keelteoskus

inglise keel	väga hea
eesti keel	kesktase
saksa keel	algtase
araabia keel	emakeel

Teenistuskäik

01.09.2022–...	Tallinna Tehnikaülikool, Inseneriteaduskond, Elektrienergia ja mehhatroonika instituut, nooremteadur (1,00)
01.01.2021–30.09.2021	Egiptuse-Vene Ülikool, Inseneriteaduskond, Telekommunikatsiooni ja elektroonika osakond, õppejõu assistent (1,00)
01.10.2018–31.12.2020	Egiptuse-Vene Ülikool, Inseneriteaduskond, Telekommunikatsiooni ja elektroonika osakond, õpetamisassistent (1,00)

ISSN 2585-6901 (PDF)
ISBN 978-9916-80-294-6 (PDF)