Towards Next-Generation Intelligent and Secure Electric Drive Systems

by

Bowen Yang

(Under the Direction of Jin Ye)

Abstract

This dissertation addresses the growing significance of cybersecurity in electric drive systems with applications on power systems, electric vehicles, and intelligent manufacturing systems as these domains increasingly rely on digital technologies and interconnectivity. This dissertation presents a comprehensive approach to tackle cybersecurity challenges in three key areas: (1) cyber-physical modeling of intelligent electric drive systems, (2) development of a cyber-physical testbed for modern industrial motor drives, and (3) implementation of advanced anomaly detection and root-cause diagnosis algorithms. The proposed methodology for modeling intelligent electric drive systems integrates physics-equation-based and control information flow models, creating an analytical framework that encompasses both physical faults and cyberattacks. This framework bridges the gaps between current modeling methods employed by different communities and facilitates seamless connections between existing modeling approaches from the cyber and physical domains. The development of a cyber-physical testbed for modern industrial motor drives includes a hardware-in-the-loop (HIL) real-time simulation testbed and a lab-scale real-world hardware experiment testbed. The HIL testbed mitigates potential risks and costs during the research and development process, while the lab-scale hardware experiment testbed focuses on real-world data generation, system final validation, and prototype demonstration. Furthermore, this dissertation develops data-driven anomaly detection and root-cause diagnosis methods to address critical problems in existing monitoring systems for both cyberattacks and physical faults. The proposed methods reduce time-to-detect, achieve high diagnostic accuracy, enable monitoring of multiple motor drives with limited sensors, and decrease dependence on large amounts of high-cost and high-risk experimental data sets. In the end, a protype is built for intelligent electric drive systems to facilitate real-world security demonstrations, which integrates attack modeling and analysis, detection, and root cause diagnosis. In summary, this dissertation provides a promising direction for future research and development efforts aimed at enhancing the cybersecurity of electric drive systems in critical power systems, electric vehicles, and intelligent manufacturing systems.

Index words: [Electric Drive System, Electric Machine, Power Electronics, Cyber-Physical System, Cybersecurity, Hardware-in-the-Loop Simulation, Anomaly Detection, Root-Cause Diagnosis]

Towards Next-Generation Intelligent and Secure Electric Drive Systems

by

Bowen Yang

B.S., Huazhong University of Science and Technology, China, June 30, 2018

A Dissertation Submitted to the Graduate Faculty of the
University of Georgia in Partial Fulfillment of the Requirements for the Degree.

Doctor of Philosophy

Athens, Georgia

2023

Towards Next-Generation Intelligent and Secure Electric Drive Systems

by

Bowen Yang

Major Professor: Jin Ye

Committee: Shixin Jack Hu
Kyle Johnsen
Ping Ma
Wenzhan Song

Electronic Version Approved:

Ron Walcott
Dean of the Graduate School
The University of Georgia
May 2023

Acknowledgments

I would like to express my sincere gratitude to the following individuals, without whom this dissertation would not have been possible: First and foremost, I am grateful to my parents for their unwavering love and support throughout my academic journey. Their encouragement and sacrifices have made it possible for me to pursue my dreams. I would like to thank my major advisor, Dr. Ye, for her invaluable guidance, patience, and expertise throughout the entire process. Her mentorship has been instrumental in shaping my research and academic growth. I am also grateful to my committee members, Dr. Hu, Dr. Johnsen, Dr. Ma, and Dr. Song, for their insightful feedback, valuable suggestions, and constructive criticism. Their contributions have helped me to refine my research and enhance the quality of my dissertation. I would like to extend my appreciation to my friends and colleagues who have supported me along the way. Their encouragement, intellectual discussions, and willingness to lend a helping hand have been a source of inspiration and motivation. Last but not least, I would like to thank all of the participants who took part in my study, without whom this research would not have been possible. Thank you all for your support and encouragement throughout my academic journey. I am deeply grateful for your contributions and honored to have had the opportunity to work with such an exceptional group of individuals.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Cybersecurity has become a critical concern in modern-day power systems, electric vehicles, and intelligent manufacturing systems. As these systems become more interconnected and automated, the risk of cyber attacks and their potential impact on society increases.

Power Systems: The power grid is a complex and critical infrastructure that is susceptible to cyber attacks. A successful attack could result in widespread power outages, economic losses, and even loss of life. One potential vulnerability is the use of legacy systems that lack modern security features. These systems may be more susceptible to cyber attacks and can be difficult to update or replace. Other vulnerabilities include supply chain attacks, where attackers target the manufacturers of power system components or software, and insider threats, where individuals with access to critical systems intentionally or unintentionally cause harm. To mitigate these risks, power systems operators can implement a range of cybersecurity measures, including network segmentation, intrusion detection systems, and regular security audits. Additionally, the development of new technologies, such as blockchain, can provide secure communication and data exchange between different components of the power grid. (Cárdenas et al., 2008; Ten et al., 2010; Yan et al., 2012)

Electric Vehicles: Electric vehicles (EVs) are becoming more prevalent in society, with a growing number of models on the market and an increasing number of charging stations. However, as with any connected device, EVs are susceptible to cyber attacks. These attacks could compromise the safety and security of both the vehicle and its occupants, as well as the wider power grid. Potential vulnerabilities in EVs include the use of wireless communication systems that could be intercepted by attackers, and the use of cloud-based services that may not be adequately secured. To mitigate these risks, EV manufacturers can implement secure communication protocols and regularly update the vehicle's software to address new threats. Additionally, the development of secure charging infrastructure and the use of blockchain technology can improve the security of the wider power grid.(Bharati et al., 2020; Checkoway et al., 2011)

Intelligent Manufacturing Systems: Intelligent manufacturing systems are highly automated and connected systems that use data analytics, artificial intelligence, and the internet of things (IoT) to op-

timize production processes. However, the increased connectivity and reliance on software systems can also make these systems more vulnerable to cyber attacks. One potential vulnerability in intelligent manufacturing systems is the use of unsecured IoT devices. These devices can provide attackers with a foothold into the system and allow them to access critical data or cause physical damage. Additionally, supply chain attacks can target the manufacturers of components or software used in these systems. To mitigate these risks, manufacturers can implement a range of cybersecurity measures, including secure communication protocols, network segmentation, and regular security audits. The development of secure IoT devices and the use of blockchain technology can also improve the security of intelligent manufacturing systems.(Y. Liao et al., 2017; Wang & Wang, 2016)

Electric drive systems are one of the most critical elements in power systems, electric vehicles, and intelligent manufacturing systems. However, while most of existing cyberaecurity studies focused on system level controls and communications, little attention is given to the electric drive systems. Electric drive systems, which are used to control the speed and torque of electric motors, have become increasingly popular in various applications, including electric vehicles, industrial automation, and renewable energy systems. However, these systems also pose unique cybersecurity challenges, as they are connected to external networks and can be targeted by malicious actors. Several cybersecurity studies have been conducted on electric motor drives to identify vulnerabilities and develop strategies to mitigate potential attacks. These studies have focused on various components of electric motor drives, including the motor control unit (MCU), the power electronics, and the communication networks. One key vulnerability in electric motor drives is the MCU, which controls the operation of the motor. Researchers have identified several potential attacks on the MCU, including denial of service attacks, firmware manipulation, and physical attacks. To mitigate these attacks, researchers have proposed using secure boot mechanisms, intrusion detection systems, and hardware-based security solutions. The power electronics, which convert electrical power to the appropriate form for the motor, are also vulnerable to cyber attacks. Researchers have demonstrated the ability to disrupt the operation of the power electronics by exploiting vulnerabilities in the control algorithms or by injecting malicious signals. To prevent these types of attacks, researchers have proposed using secure communication protocols and implementing authentication and access control mechanisms. Finally, the communication networks that connect different components of electric motor drives are also vulnerable to attacks. Researchers have demonstrated the ability to intercept and modify messages sent over these networks, which could allow an attacker to control the motor or disrupt its operations. To prevent these attacks, researchers have proposed using secure communication protocols, such as Transport Layer Security (TLS), and implementing authentication and access control mechanisms.

Overall, cybersecurity studies on electric motor drives have identified several potential vulnerabilities and proposed strategies to mitigate the risk of cyber attacks. However, as electric motor drives continue to evolve and become more connected, it will be important for researchers and industry stakeholders to continue to monitor and address new and emerging threats to these systems. (Yang, Guo, Li, et al., 2020a; J. Ye et al., 2020)

1.2 Real-World Examples of Cyberattacks Targeting Intelligent Electric Drive Systems

1.2.1 Stuxnet Attacks

Stuxnet is a highly sophisticated computer worm that was first discovered in June 2010. It is considered to be the first cyber weapon that was specifically designed to target industrial control systems (ICS). The worm was initially designed to target and disrupt the Iranian nuclear program, and its discovery brought to light the potential impact of cyber attacks on critical infrastructure. Stuxnet was designed to target Siemens programmable logic controllers (PLCs) that were used in centrifuges at Iran's Natanz nuclear facility. The worm would infect the PLCs and modify the code that controlled the speed of the centrifuges, causing them to spin out of control and eventually fail. Stuxnet was able to spread by exploiting several zero-day vulnerabilities in Microsoft Windows operating systems, and it was able to propagate through USB drives. The Stuxnet attack was highly sophisticated and demonstrated a level of expertise and resources that were previously unseen in cyber attacks. The worm was designed to evade detection and analysis by security researchers, and it contained several layers of encryption and obfuscation. It was also able to self-replicate and update itself as it spread through different systems. The discovery of Stuxnet highlighted the potential impact of cyber attacks on critical infrastructure and raised concerns about the security of industrial control systems. It also highlighted the need for greater collaboration between governments, the private sector, and security researchers to address the growing threat of cyber attacks. In response to the Stuxnet attack, governments and the private sector have taken steps to improve the security of critical infrastructure. This includes increased investment in cybersecurity research and development, the development of standards and guidelines for securing industrial control systems, and the establishment of information-sharing networks to enable the exchange of threat intelligence.

In summary, the Stuxnet attack was a highly sophisticated and targeted cyber attack that demonstrated the potential impact of cyber attacks on critical infrastructure. While the attack was targeted at Iran's nuclear program, it highlighted the need for greater security measures to protect industrial control systems around the world. The response to the Stuxnet attack has led to increased collaboration and investment in cybersecurity, which will hopefully lead to more secure and resilient critical infrastructure in the future. (Karnouskos, 2011; Langner, 2011; Zetter, 2011)

1.2.2 Jeep Cherokee Hack in 2015

In 2015, a group of researchers demonstrated a remote exploit of a Jeep Cherokee's vehicle systems through its vulnerable infotainment system. The hack was accomplished through the exploitation of the radio head unit's firmware, allowing attackers to remotely control various systems within the vehicle, including the engine, transmission, and brakes. The attack demonstrated the potential for serious consequences, including loss of life, resulting from cyberattacks on vehicle systems. This work provides an analysis of the Jeep Cherokee hack, including the vulnerabilities that allowed the attack to be carried out, the techniques

used by the attackers, and the potential consequences of such an attack. The Jeep Cherokee hack was accomplished through the exploitation of several vulnerabilities within the vehicle's infotainment system. These vulnerabilities included the use of an outdated version of the infotainment system's software, which was known to contain security vulnerabilities, and the lack of adequate security measures to protect against remote attacks. Additionally, the infotainment system was connected to the internet, allowing attackers to remotely access the system and exploit its vulnerabilities. The Jeep Cherokee hack was accomplished through the use of several techniques, including the use of a cellular network to remotely access the vehicle's infotainment system, the exploitation of a vulnerability in the radio head unit's firmware, and the use of reverse engineering to identify the vehicle's control systems. Once the attackers had gained access to the vehicle's systems, they were able to remotely control various systems within the vehicle, including the engine, transmission, and brakes. The potential consequences of the Jeep Cherokee hack were significant, including loss of life resulting from a cyberattack on a vehicle's systems. While the researchers who carried out the attack did not intend to cause harm, the attack demonstrated the potential for serious consequences resulting from cyberattacks on vehicle systems. Additionally, the attack highlighted the need for enhanced security measures to protect against similar attacks in the future.

The Jeep Cherokee hack was a significant demonstration of the potential consequences of cyberattacks on vehicle systems. The vulnerabilities that allowed the attack to be carried out, the techniques used by the attackers, and the potential consequences of such an attack highlight the importance of secure software development practices and the need for enhanced security measures to protect against similar attacks in the future. (Miller & Valasek, 2015)

1.2.3 Ukrainian Power Grid Attack in 2015

In December 2015, a coordinated cyberattack targeted the Ukrainian power grid, resulting in a widespread blackout that left over 200,000 customers without power for several hours. The attack was carried out through a combination of sophisticated malware and social engineering tactics, highlighting the potential for significant consequences resulting from cyberattacks on critical infrastructure. This work provides an analysis of the Ukrainian power grid attack, including the tactics used by the attackers, the vulnerabilities that allowed the attack to be carried out, and the lessons learned from the attack. The Ukrainian power grid attack was carried out through a combination of sophisticated malware and social engineering tactics. The attackers gained access to the power grid's control systems through spear-phishing attacks, which involved sending emails with malicious attachments to employees of the power grid. Once inside the system, the attackers used malware to gain control of critical systems, including the circuit breakers and other devices that control the flow of power. The attackers then remotely triggered the circuit breakers, causing a widespread blackout. The Ukrainian power grid attack exploited several vulnerabilities within the power grid's control systems, including the use of outdated software and weak passwords. Additionally, the attackers were able to gain access to the system through spear-phishing attacks, highlighting the importance of employee awareness and training in preventing cyberattacks on critical infrastructure. The Ukrainian power grid attack highlighted several lessons learned in protecting critical infrastructure against cyberattacks. These include the need for enhanced security measures, including the use of up-

to-date software and strong passwords, as well as the importance of employee awareness and training in preventing cyberattacks. Additionally, the attack highlighted the need for international cooperation in protecting against similar attacks, as cyberattacks on critical infrastructure are a global threat that requires a coordinated response.

The Ukrainian power grid attack was a significant demonstration of the potential consequences of cyberattacks on critical infrastructure. The tactics used by the attackers, the vulnerabilities that allowed the attack to be carried out, and the lessons learned from the attack highlight the need for enhanced security measures and the importance of international cooperation in protecting against similar attacks in the future. (Koppel, 2016)

1.3 Dissertation Outlines

According to previous discussions, intelligent electric drive systems, one of the most critical elements of power systems, electric vehicles, and manufacturing systems, require more thorough studies concerning cybersecurity issues. While existing cybersecurity studies primarily come from computer science communities and focus on communication protocols and software architectures, more efforts have yet to be made from electrical engineering perspectives. As intelligent electric drive systems are essentially cyber-physical systems, physical-domain perspectives should also preserve great potential regarding cyber security problems. So far, existing electrical engineering literature has regarded physical fault research as the primary direction regarding system safety and security. Therefore, this work tries to fill this gap by extending traditional electric drive system physical fault analysis to cover cybersecurity aspects.

Fig. 1.1 shows a conceptual diagram of the dissertation outlines. This dissertation approaches the above research question from system modeling, testbed platform, and monitoring algorithms.

Chapter 2 will first discuss the empirical impact analysis of various cyberattacks on intelligent electric drive systems. Then, it will elaborate on the proposed advanced cyber-physical modeling approach for intelligent electric drive systems. Such a modeling method originates from physics-equation-based modeling methods and combines a novel control information flow model for cyberattacks. Detailed derivations and validation results will be provided as support for the proposed modeling approach.

Chapter 3 will describe the development of two advanced testbed platforms for cyber-physical security and safety studies. The first platform is a hardware-in-the-loop real-time simulation platform formed by OPAL-RT real-time simulator and TI C2000 microcontrollers. The real-time simulator will simulate the complex physical systems in real-time, and the TI C2000 microcontrollers will implement electric drive control algorithms and emulate various cyberattack scenarios. The second platform is a real-world hardware experiment platform consisting of two 1.5kW induction machine drives and two 1.5kW permanent magnet synchronous machine drives. These two platforms will be the fundamentals of system analysis, data generation, algorithm validations, and prototype demonstrations.

Chapter 4 will discuss four developed anomaly detection and root-cause diagnostic methods. The first method uses time-domain features alongside data-driven classifiers to achieve fast detection. The second method uses frequency-domain features to distinguish stealthy attacks from common physical faults. The

third method tries to realize accurate detection and diagnosis results with limited available sensors. The last method combines convolutional neural networks and transfer learning to ease the dependency on a large amount of experimental data sets. All these methods are tested and validated by the testbed platforms in chapter 3.

Chapter 5 will showcase a prototype monitoring solution combining findings and developments from previous chapters.

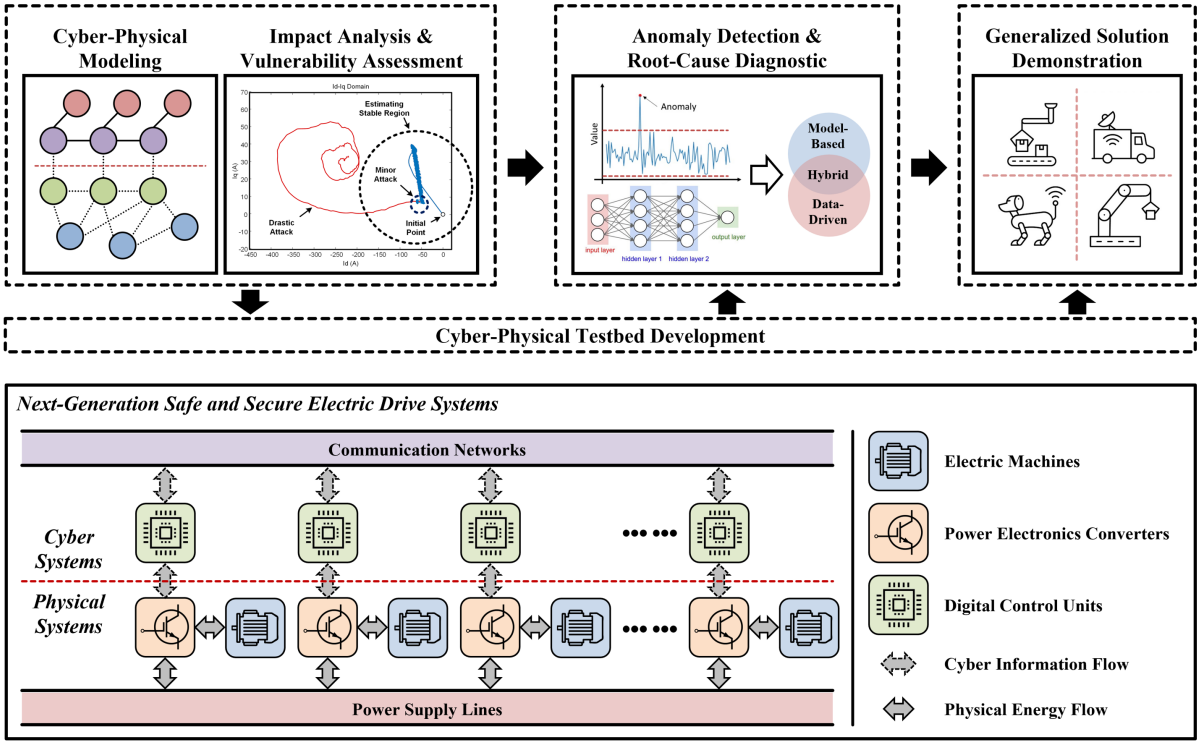


Figure 1.1: Conceptual diagram of the dissertation outlines.

Chapter 2

Cyber-Physical Modeling and Impact Analysis of Intelligent Electric Drive Systems

2.1 Impact Analysis of Cyberattacks on Electric Drive Systems

Recent years have witnessed a significant development in cyber-physical systems, which has permeated modern industrial systems, including energy production, power electronics, manufacturing, and automotive industry (Pasqualetti et al., 2015). However, due to a large number of communications and complex networks, it also brings cybersecurity concerns (F. Li, Shi, et al., 2019; F. Li, Shinde, et al., 2019). Especially for the electric systems (like power grids, wind farms and electric vehicles), as huge amount of energy contained in the power equipment is fully controlled by networked electronic units, the systems are directly exposed to the cyber threats. Once the attackers have compromised any of the controllers without being detected, catastrophic damages are usually inevitable. For instance, on August 14, 2003, a large scale electric power blackout occurred in North America which was caused by a software program failure in the power system, and this events affected around 50 million people and 61,800 megawatts of electric loads (*Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations*, 2004). In 2010, a computer worm 'Stuxnet' was discovered targeting Siemens industrial software and equipment to unstable the power system operation (Cherry & Langner, 2010; Karnouskos, 2011; McMillan, 2010). In addition to power systems, electric drive systems, which are playing an increasingly important role in industrial applications, can also be attacked through maliciously modifying controller in real-life applications. In 2015, an unnamed steel mill in Germany was attacked. The hackers were reported to maliciously manipulate the control system such that a blast furnace could not be properly shut down, resulting in unspecified "massive" damage. This is the second confirmed case of a cyber-physical attack that caused physical destruction of equipment after Stuxnet (Zetter, 2015). Recently, in March 2019, hackers in Tencent Keen Security Lab attacked Tesla's autopilot and manipulated the control of the vehicle that is powered by electric drive systems. This cyber-attack can have serious

consequences such as causing the electric vehicle to suddenly switching lanes (Lab, 2019). In August 2019, security researchers found a zero-day vulnerability in a popular building controller used for managing various systems, including HVAC (heating, ventilation, and air conditioning) (Ilascu, 2019), and they were able to maliciously modify the controller such as through modifying sensors (for instance, temperature sensors). This will later adversely impact HVAC systems, which are primarily electric drive systems.

From these real-world events, concerns of such vulnerabilities regarding cyberattacks in modern electric systems have caught research interests from various societies. Different communities have started to address safety and security problems in modern electric drive systems, including electric vehicles (J. Ye et al., 2021), solar farms (J. Ye et al., 2022), multi-agent systems (D. Zhang et al., 2021), and other cyber-physical systems in general (Humayed et al., 2017). In the field of smart grids, (Sridhar & Manimaran, 2010) analyzed the data integrity attacks on automatic generation control loop. In (Phillips et al., 2005), the cyber security policies for flexible alternating current transmission devices are discussed. In (Berthier et al., 2010; Valenzuela et al., 2013), intrusion detection methods are proposed for advanced metering infrastructures and power system operations. In (Subasi et al., 2018; Zhou et al., 2018), advanced data processing approaches (e.g. data mining and neural network) are used to detect and classify different cyber attacks targeting on smart grids. With regard to vulnerability assessment, (Xie et al., 2011) presented the impact of integrity attacks on electric market operations; ("Cyber attack in a two-area power system: Impact identification using reachability", 2010) used reachability methods in graph theory to assess the risks and vulnerabilities of two-area power systems; (Ten et al., 2010) proposed the attack and defence modeling for critical cyber infrastructures. In addition, in (Cárdenas et al., 2011; Huang et al., 2009; Kiss et al., 2015), methods of assessment, detection and possible countermeasures for process control systems have been researched. However, while cyber security studies in smart grids and critical infrastructure mostly focused on system-level, little work has been developed for device-level cyber security in electric drive systems, which are an important and vulnerable part of industrial environments (e.g. electric vehicle, intelligent manufacturing, renewable energy and smart city). Although some methods for smart grids can be effective in addressing different types of cyber attacks, they can hardly be applied to electric drive systems directly. The reasons are as follows:

1. Existing resilient control methodologies for smart grids mainly focus on few metrics such as active (or reactive) power, system frequency, node voltage, and power angle, which may be unfeasible for electric drive systems;

2. The power equipment (like generators, motors, transformers and transmission lines) are often modeled as voltage sources, impedance, or electric power loads in power grids to simplify vulnerability assessment since only stability, efficiency and economic performances of the grid are of concern. However, in electric drive systems, more detailed models (device and system) and different metrics should be considered to evaluate the system comprehensively. For example, an electric vehicle requires fast and accurate tracking of the torque and speed references to ensure dynamic performance, low torque ripple to reduce mechanic vibrations and noise, low current total harmonics distortion (THD) and high power factor to extend the life cycle of battery packs, as well as minimizing power losses to enhance the driving range.

This section intend to provide an emperical impact analysis of cyberattacks on electric drive systems as a basis for better understandings of safety and security problems for modern intelligent electric drive systems. Fig. 2.1 shows a general diagram of the electric drive systems with a large number of information exchanges between sensors and local/higher controllers, which are vulnerable to cyber and physical threats. In Fig. 2.1, attack vectors denoted in red represent potential cyber-attacks on electric drive systems. For example, local sensor signals could be modified or blocked to make the system unstable (attack A); or control signals from higher level controllers could be delayed or fabricated to lower the system efficiency (attack C); or even more malicious attacks could target on the switching signals to make power modules nonfunctional (attack D), etc. Based on this general structure, this section proposes: (1) novel evaluation metrics of electric drive systems for evaluating the system condition under a variety of sensor data integrity attacks, including more realistic and sophisticated attacks; (2) innovative index-based resilience and security criteria specifically for electric drive systems; (3) qualitative attack impact analysis on the dynamic performance and quantitative analysis on the impact index due to different cyber attacks.

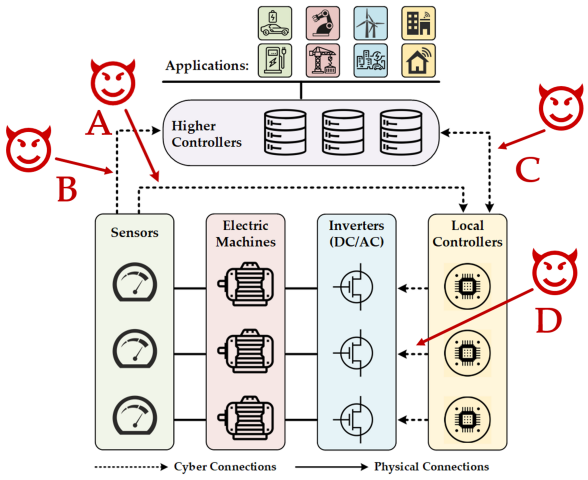


Figure 2.1: General diagram of the intelligent electric drive systems.

2.1.1 Example System Descriptions: PMSM Based Electric Drive Systems

The impact analysis is based on an electric drive system with a pamanent magnet synchronous machine (PMSM). Fig. 2.2 shows a diagram of the system physical layout and digital control logics. In the physical layer, a DC power source form a DC bus, which can provide the power input to the motor inverter. Then, the inverter drives the PMSM under the control of the PWM signals, which are generated by the local controller ($S_1 - S_6$). Sensors are implemented on the inverter output ports and collect three phase currents from the electric machine windings. The cyber layer is mainly in charge of receiving and processing the sensor signals, generating the PWM control signals, and communicating with external devices.

PMSM is currently widely adopted in applications like electric vehicles and servo systems. Under the traditional three phase static reference frame, the electrical relationships in each phase could be described as

$$\boldsymbol{v} = \boldsymbol{R}\boldsymbol{i} + \frac{d}{dt}\boldsymbol{\lambda},\ \boldsymbol{\lambda} = \boldsymbol{L}\boldsymbol{i},\ \boldsymbol{L} = \begin{bmatrix} L_{aa} & L_{ab} & L_{ac} & L_{af} \\ L_{ba} & L_{bb} & L_{bc} & L_{bf} \\ L_{ca} & L_{cb} & L_{cc} & L_{cf} \\ L_{fa} & L_{fb} & L_{fc} & L_{ff} \end{bmatrix} \tag{2.1}$$

where:

- $\boldsymbol{v} = [v_a, v_b, v_c, v_f]^T$
- $\boldsymbol{i} = [i_a, i_b, i_c, i_f]^T$
- $\boldsymbol{\lambda} = [\lambda_a, \lambda_b, \lambda_c, \lambda_f]^T$
- $\boldsymbol{R} = diag[R_a, R_b, R_c, R_f]$

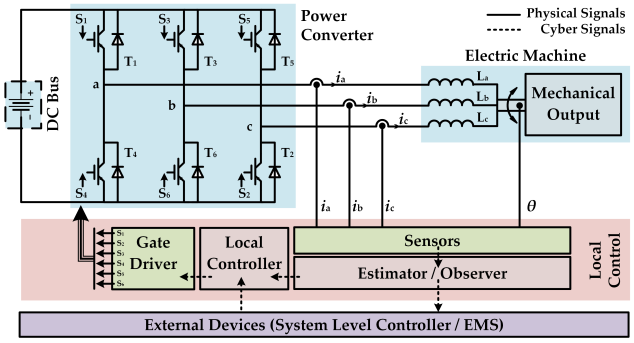


Figure 2.2: Diagram of physical layout and digital control logics of the example electric drive system.

More specifically, the flux linkage $\lambda_f = \lambda_{pm}$ is produced by the magnet mounted in the rotor; v_f, i_f, R_f represent the equivalent excitation voltage, current and resistance, respectively; L_{fx} and $L_{xf}, x = a, b, c$ reflect the flux linkage in each phase provided by rotor magnet. To simplify the analysis, Direct-Quadrant-Zero (DQ) transformation is adopted to transfer the variables in the stator static reference frame to the rotor rotating reference frame. Then the results could be described as

1. Flux Linkage:

$$\begin{cases} \lambda_d = L_d i_d + \lambda_{pm} \\ \lambda_q = L_q i_q \end{cases} \tag{2.2}$$

2. Voltage:

$$\begin{cases} v_d = R_s i_d + L_d \frac{di_d}{dt} - \omega_e L_q i_q \\ v_q = R_s i_q + L_q \frac{di_q}{dt} + \omega_e L_d i_d + \omega_e \lambda_{pm} \end{cases} \tag{2.3}$$

3. Torque:

$$T_e = \frac{3}{2} p[\lambda_{pm} i_q + (L_d - L_q) i_d i_q] \tag{2.4}$$

where L_d and L_q are the inductance of d-axis and q-axis; ω_e is the electrical angular speed; p is number of pole pairs; and R_s is the equivalent winding resistance in the DQ reference frame. It should be noted that when the stator winding is connected in 'Y' model, the zero component will always be o as suggested by Kirchhoff's Law. That is the reason why zero component is not included in the DQ model described above. Additionally, the motor speed, D-axis and Q-axis current are controlled by three Proportional-Integral (PI) regulators, which are one of the most widely used controllers in the industrial environment. Meanwhile, the reference current vector $[i_d, i_q]^T$ is selected to track the torque command. To achieve the maximum system efficiency while producing the same torque, a widely implemented algorithm, Maximum-Torque-Per-Ampere (MTPA), is adopted to optimize the current vectors. The diagram of the optimization is shown in Fig. 2.3, where the blue circle denotes the current limits and the red ellipse is the voltage limits; the yellow and green trajectories are MTPA and Maximum-Torque-Per-Voltage (MTPV), respectively; the purple trajectory defines the constant torque profile. Detailed procedures of the algorithm are described below:

1. If the system is operating with a speed lower than the basic speed ω_b, the reference current vector should follow the MTPA trajectory defined by

$$i_{dref} = \frac{\lambda_{pm}}{2(L_q - L_d)} - \sqrt{\frac{\lambda_{pm}^2}{4(L_q - L_d)^2} + i_{qref}^2} \tag{2.5}$$

2. If the speed is higher than ω_b but lower than ω_2, the reference current vector should follow MTPA trajectory while the vector is inside the voltage limit. Otherwise, if the vector is outside the voltage limit, such as point E, the current vector should be selected by the Flux Weakening Control (FWC) as

$$i_{dref} = -\frac{\lambda_{pm}}{L_d} + \frac{1}{L_d}\sqrt{(\frac{V_{smax}}{\omega_e})^2 - (L_q i_{qref})^2} \tag{2.6}$$

   denoted as point D in the figure.

3. If the speed is higher than ω_2, the reference current vector should follow the MTPA trajectory while the vector is inside the voltage limit ellipse. Otherwise, the current vector should follow the MTPV trajectory defined by

$$(L_d - L_q)[(\frac{L_d i_{dref} + \lambda_{pm}}{L_q})^2 - i_{qref}^2] + \lambda_{pm}(\frac{L_d i_{dref} + \lambda_{pm}}{L_q}) = 0 \tag{2.7}$$

Generally speaking, the grey region in Fig. 2.3 (**OABC**) is where the optimal reference current vector should be selected.

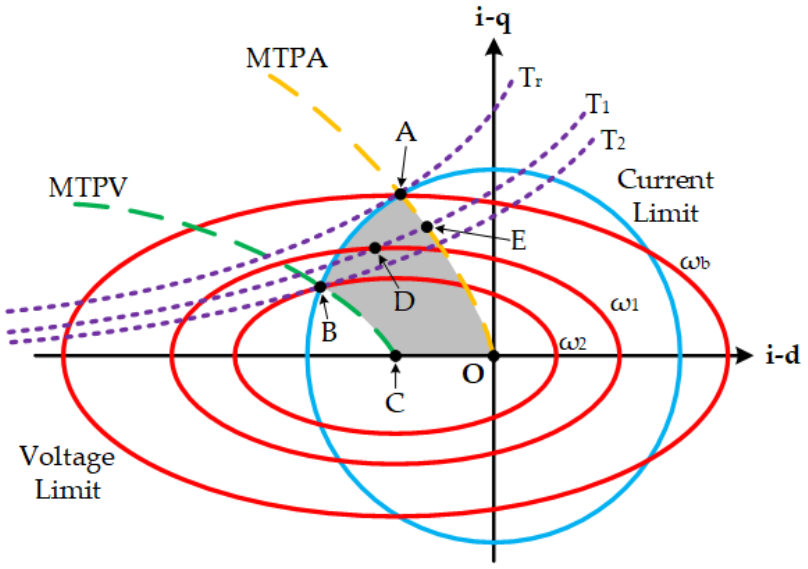


Figure 2.3: Diagram of the reference current vectors with MTPA and MTPV.

2.1.2 Cyberattack Case Study

To quantitatively analyze the impact of cyber attacks on the target electric drive systems, we suppose the real and fake feedback measurements are denoted as y and $\hat{y}$, respectively, and the time horizon under attack is $\boldsymbol{T_{ATK}} = [t_0, t_0 + T_a]$. Then, two common attacks are modeled as

$$\hat{y} = \begin{cases} y & (t \notin \boldsymbol{T_{ATK}}) \\ \alpha \cdot y & (t \in \boldsymbol{T_{ATK}}) \end{cases} \tag{2.8}$$

$$\hat{y} = \begin{cases} y & (t \notin \boldsymbol{T_{ATK}}) \\ y + \beta & (t \in \boldsymbol{T_{ATK}}) \end{cases} \tag{2.9}$$

where t_0 is the start time of the attack, and T_a is the time of attack duration. In the above attack model, α could be greater than 1, meaning the signal is falsely amplified, or smaller than 1, meaning the signal is falsely reduced; β could be a constant or a complex function. Here, β is modeled as three different functions: a white noise injection, a decaying high frequency harmonics injection, and a periodic pulse injection, which can be expressed as follows:

$$\beta = \text{white noise} \tag{2.10}$$

which is defined by the energy and sampling time;

$$\beta = Ae^{-t/\tau} \cdot \sin(2\pi \cdot f \cdot t) \tag{2.11}$$

where A, τ, and f represent the oscillation amplitude, decaying coefficient, and oscillation frequency, respectively;

$$\beta = \mathscr{F}(t) = \begin{cases} K & (kT_s \leqslant t < D \cdot T_s + kT_s) \\ 0 & (D \cdot T_s + kT_s \leqslant t < (k+1)T_s) \end{cases} \tag{2.12}$$

where k is an integer, and D, T_s, K are the duty cycle, signal period and attack amplitude, respectively. Based on such attack models, five specific types attacks are established targeting on single phase, two phases and three phases respectively, which results in fifteen cases as shown in Table 2.1. It should be pointed out that all these 15 cases are not meticulously designed for some specific attack purpose. Nevertheless, they could be used as preliminary demonstrations

Table 2.1: Attack Modeling and Case Definition

Case Definition		Attack Targets		
		Phase A	Phase A and B	Phase A, B and C
$\hat{y} = \alpha y$	$\alpha = 0.8$ (type 1)	Case 1	Case 6	Case 11
	$\alpha = 1.2$ (type 2)	Case 2	Case 7	Case 12
$\hat{y} = y + \beta$	$\beta = noise$ (type 3)	Case 3	Case 8	Case 13
	$\beta = 25e^{-t/0.1} \cdot \sin(2\pi \cdot 200 \cdot t)$ (type 4)	Case 4	Case 9	Case 14
	$\beta = \mathscr{F}(t)$ where $K = 30, Ts = 0.001, D = 0.25$ (type 5)	Case 5	Case 10	Case 15

2.1.3 Evaluation Metrics

In order to provide an insight into the impact of cyberattacks on the system, as well as to give a guideline for developing monitoring and detection methodology, we propose to use a series of new metrics below. To obtain the transient process of the attack, all metrics are calculated within a sliding window, denoted as $\mathscr{T}_w$.

Torque Ripple and Speed Ripple

The torque and speed ripple (marked as S_1 and S_2, respectively) reflect the mechanical characteristics, which are of vital importance to an electric drive system. Large torque or speed ripple can normally bring damages to the mechanical components such as rotor and shaft, and can lead to other negative consequences like drive performance discomfort in electric vehicles and poor motion accuracy in manufacturing and servo systems. The two indices are defined as

$$\begin{aligned} S_1(t_0) &= \frac{max\{T(t)\} - min\{T(t)\}}{ave\{T(t)\}} \\ S_2(t_0) &= \frac{max\{n(t)\} - min\{n(t)\}}{ave\{n(t)\}} \end{aligned} \tag{2.13}$$

where $t \in [t_0, t_0 + \mathscr{T}_w]$; T and n are the electromagnetic torque and the rotating speed.

Current Distortion Index

The current distortion index S_3 is defined to show the degree of current distortion caused by the attack, expressed as

$$S_3 = \sqrt{\frac{\int_{-\infty}^{f_l} I^2(f)df + \int_{f_u}^{+\infty} I^2(f)df}{\int_{f_l}^{f_u} I^2(f)df}} \tag{2.14}$$

where $I(f)$ is the amplitude of the phase current in frequency domain after the Fourier Transformation. As harmonics apart from operating frequency may lead to damage to hardware devices such as battery packs and IGBTs in the inverter, the distortion should be maintained as low as possible. The dominant current frequency is found to fluctuate around the operating point, and thus a frequency band $[f_l, f_u]$ is introduced here to differentiate the normal frequency fluctuation and the current distortion caused by attacks,defined by

$$f_u = f_0 + \Delta f,\ f_l = f_0 - \Delta f. \tag{2.15}$$

Here f_0 is the operating frequency calculated from the average speed, and Δf denotes the bandwidth.

Torque Tracking Error

The torque tracking error, due to the ability to depict the dynamic response characteristic, is normally defined to measure the torque tracking performance and determine whether the system is working at desired operating point. It is defined by

$$S_4 = \frac{\sqrt{\frac{1}{\mathscr{T}_w}\int_{t_0}^{t_0+\mathscr{T}_w}(T_{ref}(t)-T_e(t))^2 dt}}{ave\{T_{ref}(t)\}} \tag{2.16}$$

where T_{ref} is torque reference; T_e denotes actual torque which could be directly measured or calculated by the phase current.

Three Phase Current Unbalance Components

The three phase current unbalance components is calculated by the asymmetric components methods, which reflects the asymmetry features among three phase currents. The detailed calculating procedure is shown in Fig. 2.4, wherein, $[\mathbf{N-Park}]$ is the Park transformation matrix in the negative sequence; LPF represents low pass filter and is derived by

$$G(s) = \frac{(2\pi \cdot f)^2}{(s+2\pi \cdot f)^2} \tag{2.17}$$

where f is the cut-off frequency. Then, the index S_5 is the amplitude of the unbalance components, calculated by

$$S_5 = \sqrt{a^2+b^2} \tag{2.18}$$

The unbalance current will bring a great damage to the system, even leading to instability. Normally, if the system is on healthy condition, the unbalance components are approximately zero; if not, there probably exist some attacks or faults.

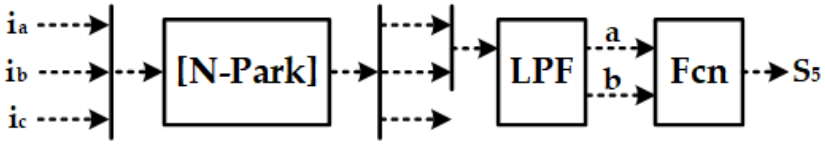


Figure 2.4: Calculation procedure of S_5.

Impact Index

As these five metrics describe different characteristics of the electric drive system respectively, a more general impact index $\mathcal{K}_{imp}$ is proposed for the purpose of comprehensive assessment. The calculation of

$\mathcal{K}_{imp}$ is shown as

$$\mathcal{K}_{Imp} = \sum_{x=1}^{5} k_{imp}^{S_x}, \tag{2.19}$$

where $k_{imp}^{S_x}$ is the impact factor for each evaluation metrics. The calculation expression is shown as

$$k_{imp}^{S_x} = (\frac{k_x^{attack} + k_x^{after-attack}}{k_x^{normal}} - 2), \tag{2.20}$$

where k_x^{attack}, $k_x^{after-attack}$, k_x^{normal} are the root-mean-square of $S_1 - S_5$ during the attack period, beyond attack period and the normal period. The detail calculation process are shown as Eq. 2.21, where $\mathcal{T}_{attack}, \mathcal{T}_{after-attack}, \mathcal{T}_{normal}$ are the different time periods.

$$\begin{aligned} k_x^{attack} &= \sqrt{\frac{1}{\mathcal{T}_{attack}} \int_{\mathcal{T}_{attack}} S_x^2 dt}, \\ k_x^{normal} &= \sqrt{\frac{1}{\mathcal{T}_{normal}} \int_{\mathcal{T}_{normal}} S_x^2 dt}. \\ k_x^{after-attack} &= \sqrt{\frac{1}{\mathcal{T}_{after-attack}} \int_{\mathcal{T}_{after-attack}} S_x^2 dt}. \end{aligned} \tag{2.21}$$

2.1.4 Simulation Results and Impact Analysis

To analyze the impact of the attack cases in Table 2.1, a 50kW PMSM-based electric drive system is built in MATLAB Simulink with the same hardware topology and controller diagram shown in Fig. 2.2. Three main works are presented here: 1) basing on the defined evaluation metrics and control theory, we propose an analytic methodology of evaluating system stability, security and resilience, and the metric-based boundaries can be used further to detect the malicious attacks online; 2) qualitative and quantitative impact of attacks are analyzed in detail, and then general guidelines are summarized; 3) the statistical graph is shown in the last part, under which, we evaluate the potential damage and influence of different types of attacks on the defined metrics, which can serve as guidelines for attack detection and countermeasures in real-life applications.

Stability, Security, and Resilience of the System

Stability of the system: As the system is a high order nonlinear system, it is hard to find a proper Lyapunov function to establish the stability criteria. Therefore, to illustrate the system stability, we propose a proposition in a broad sense.

Proposition 1. *Given a continuous state space $\boldsymbol{X} \subseteq \mathbb{R}^n$, if the system has an equilibrium point $\boldsymbol{X_e}$, and there exists a set $\mathbb{B}^n$, which satisfies: 1) $\mathbb{B}^n \subseteq \mathbb{R}^n$; 2) $\boldsymbol{X_e} \subseteq \mathbb{B}^n$; 3) for any initial point $\boldsymbol{X_0} \subseteq \mathbb{B}^n$, the state*

space $\boldsymbol{X}$ will eventually converge to the equilibrium point $\boldsymbol{X_e}$. Then, the system is stable in $\mathbb{B}^n$, and $\mathbb{B}^n$ is defined as stable region of the system around the equilibrium point $\boldsymbol{X_e}$.

Fig. 2.5a shows the 2-D phase portrait (i_d, i_q) of the system. As long as the state space trajectory of the system belongs $\mathbb{B}^n$ during the attack, the system will be stable. As shown in Fig 2.5a, when a minor attack occurs, the deviation from the initial point is quite small and the whole phase trajectory is inside the boundary $\mathbb{B}^n$, the system is stable; however, if the attack is drastic, as the red trajectory, the operating point will go beyond $\mathbb{B}^n$, thus the system becomes unstable.

Security of the system: To evaluate the security of the system, we define metric-based boundaries, as

Proposition 2. *Define S_m, $m = 1, 2...$ as system evaluation metrics. If a boundary $\boldsymbol{K}_m = [k^m_{lower}, k^m_{upper}]$ could be found, which has the following properties: (1) k^m_{lower} and k^m_{upper} is finite; (2) if $S_m \in \boldsymbol{K}_m$, the damage caused by the attacks are acceptable. Then, the system is secure.*

Fig. 2.5b shows the index S_1 under two cyber attacks. Although the index under $\boldsymbol{ATK-1}$ can come to the equilibrium range after attack is removed, during the dynamic process, the damage caused by the attacks are not acceptable ($S_1 \notin \boldsymbol{K}_1$) and thus the system is not secure.

Resilience of the system: The resilience refers to the ability of recovery after suffering from malicious attacks. We consider the recovery time $T_{r,m}$ of the mth index S_m ($m = 1, 2, ...$) from the time when the attack is withdrawn to the time when the index restores to its original value. Then, the boundary reflecting the resilience is defined as:

Proposition 3. *Define $T_{r,m}$, $m = 1, 2, ...$ as the system evaluation metrics. If a boundary $\boldsymbol{T_{r,m}} = [T^{r,m}_{lower}, T^{r,m}_{upper}]$ could be found, which has the following properties: (1) $T^{r,m}_{lower}$ and $T^{r,m}_{upper}$ is finite; (2) as long as $T_{r,m} \in \boldsymbol{T_{r,m}}$, $T_{r,m}$ could restore to its original value when the attack is withdrawn. Then, the system is resilient.*

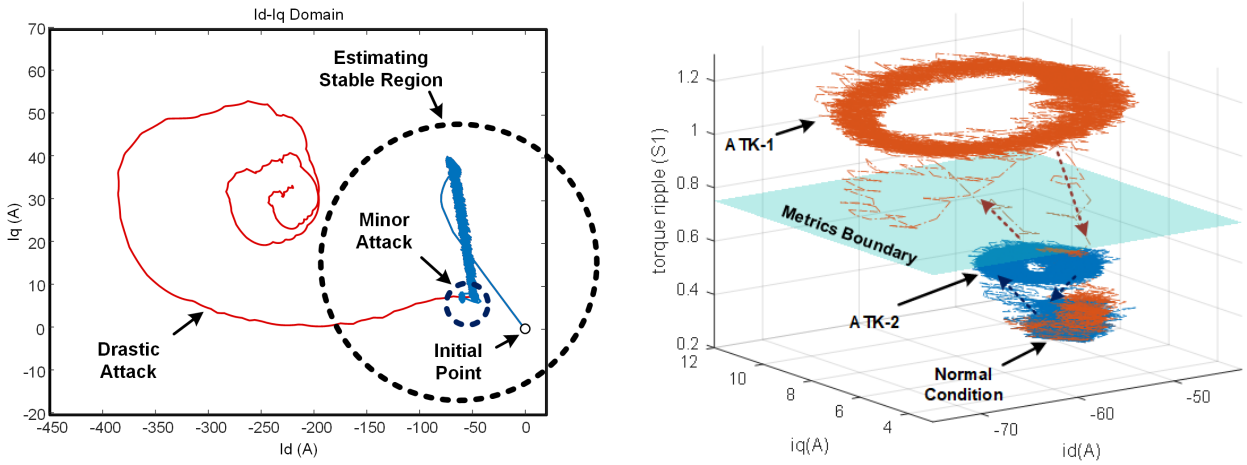
This boundary represents the resilience performance of the studied electric drive system. The larger boundary demonstrate the better resilience of the systems against cyber attacks.

Remarks: The metric-based boundaries could be obtained through massive simulations or experiments, and may vary with different application scenarios. In this subsection, we suppose the electric drive system is applied to a four-wheel driven electric vehicles. Then, the torque ripple (S_1) boundary could be selected as $[0, 0.2]$ to avoid destroying the yaw stability.

Simulation Results Under Sensor Attacks

Based on the evaluation metrics and index introduced in section III, 15 caces in Table 2.1 are simulated and analyzed. Table 2.2 shows the detail results of the impact index for each cases, which is calculated through Eq. 2.19-Eq. 2.21. Among these results in Table 2.2, the ones in case 13 - case 15 are strangely identical. The reason is that in these three cases, sensors of three phases are added by the same false signals, which means all false signals will be transferred to zero-axis component after DQZ transformation. As the control algorithms only adopt d- and q-axis information, false signals in these three cases will not influence the

controller performance. Meanwhile, besides these three, we present several cases for better observation, and a sliding window is constructed on the time axis. The trajectories are plotted in Figs. 2.6-2.11. It should be pointed out that due to the simple relationship between torque and rotation speed in this model and the similarity between the profiles of S_1 and S_2, these two metrics are drawn in one figure, for the space saving purpose. Besides, current distortion S_3 is calculated from phase A current, which is always one of the attack targets in the simulation.



(a) 2D state trajectory (id & iq)

(b) 3D metrics trajectory (torque ripple)

Figure 2.5: System trajectories under cyberattacks.

Table 2.2: Detailed Simulation Results of the Impact Index

	$k_{imp}^{S_1}$	$k_{imp}^{S_2}$	$k_{imp}^{S_3}$	$k_{imp}^{S_4}$	$k_{imp}^{S_5}$	$\mathcal{K}_{Imp}$
1	1.689	14.92	-0.003	1.275	9.521	27.40
2	1.616	14.16	-0.006	0.948	7.427	24.15
3	1.023	4.375	-0.007	0.024	0.088	5.503
4	3.090	6.446	0.034	1.664	3.1781	14.41
5	11.15	31.93	0.226	4.534	3.899	51.74
6	1.995	28.12	0.005	2.014	10.865	43.01
7	1.783	25.41	-0.003	1.448	6.567	35.20
8	1.576	5.086	-0.008	0.059	0.126	6.838
9	2.757	6.263	0.006	1.612	2.663	13.30
10	10.32	29.35	0.087	4.271	3.098	47.12
11	0.471	37.49	0.005	1.940	0.350	40.26
12	0.946	36.25	0.001	1.862	-0.074	38.98
13	-0.048	-0.089	-0.012	-0.225	-0.008	-0.383
14	-0.048	-0.089	-0.012	-0.225	-0.008	-0.383
15	-0.048	-0.089	-0.012	-0.225	-0.008	-0.383

Case 1: $\hat{y} = 0.8y$, $t \in \boldsymbol{T_{ATK}}$, targeting phase A

Fig. 2.6 shows the results when the feedback signal of phase A is reduced to 80% of the original value. It can be observed that this reduction can heavily deflect the actual current from its reference. Once the current of phase A increases, the current of phase B and phase C will drop to achieve the Kirchhoff current theorem. As a consequence, the three phases become unbalanced, which is reflected by S_5 profile. Meanwhile, as the inaccuracy of the current value, torque ripple will be increased and the current distortion will be worsened. It is also worth noting that all five metrics are bounded in the whole process and when the attack is eliminated, the system performance could be restored while a transient process is required.

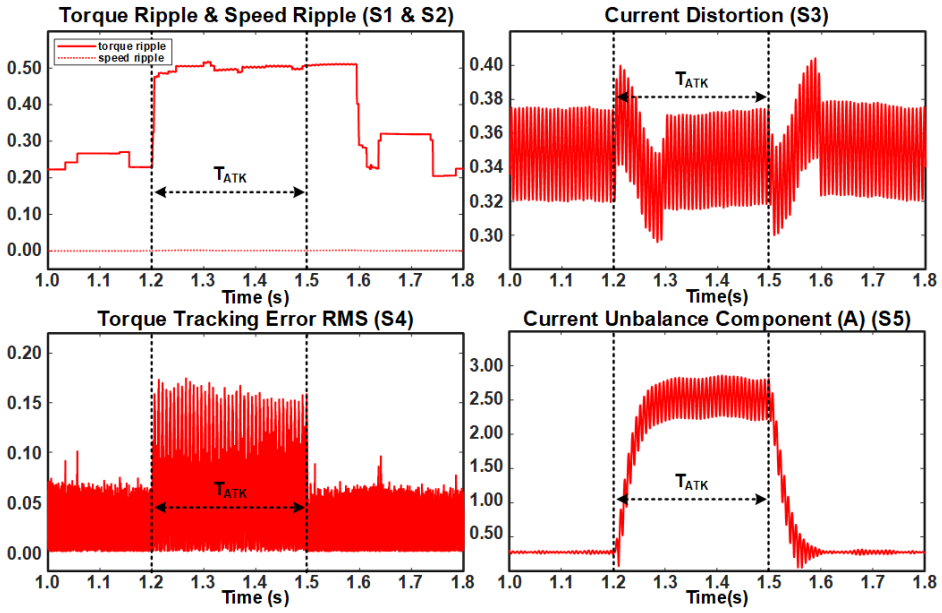


Figure 2.6: Case 1: reducing attack, $\hat{y} = 0.8y$, $t \in \boldsymbol{T_{ATK}}$, targeting phase A.

Case 2: $\hat{y} = 1.2y$, $t \in \boldsymbol{T_{ATK}}$, targeting phase A

Fig. 2.7 shows the results when phase A current sensor signal is enlarged by 1.2 times. From the results we can see that through the attack duration, the current of phase A decreases heavily, and that smaller current in phase A can lead to larger current in phase B and phase C. As the required torque cannot be provided in time, the torque controller intends to request larger current, which may cause the controller saturation, higher current distortion, tracking error and ripples. Meanwhile, we notice that the changing pattern of the current distortion S_3 is opposite between case 1 and case 2, which could be a helpful tools to distinguish increasing attacks and decreasing attacks.

Case 4: $\hat{y} = y + 25e^{-t/0.1} \cdot \sin(2\pi \cdot 200 \cdot t)$, $t \in \boldsymbol{T_{ATK}}$, targeting phase A.

Case 4 demonstrates the impact when a decaying high frequency harmonics is introduced to phase A current feedback signals. The magnitude of the harmonics is 25 ampere, the decaying coefficient is 0.1 and the oscillation frequency is 200Hz. The results are shown in Fig. 2.8. As shown in the figure, all metrics have a step change, and then a decaying change similar to the attack appeals. This feature could be an useful tool for detecting and diagnosing the decaying attacks. However, it should be noted that some

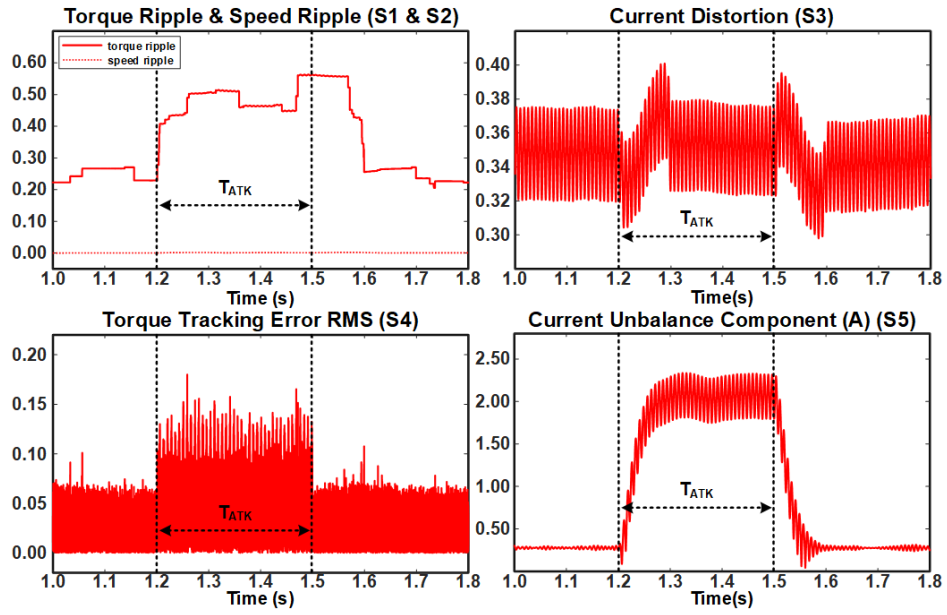


Figure 2.7: Case 2: enlarging attacks, $\hat{y} = 1.2y, t \in \boldsymbol{T_{ATK}}$, targeting phase A.

physical faults also has the decaying characteristics such as some short circuit faults. So in real application, it should be addressed with enough attention for distinguishing physical faults and malicious attacks.

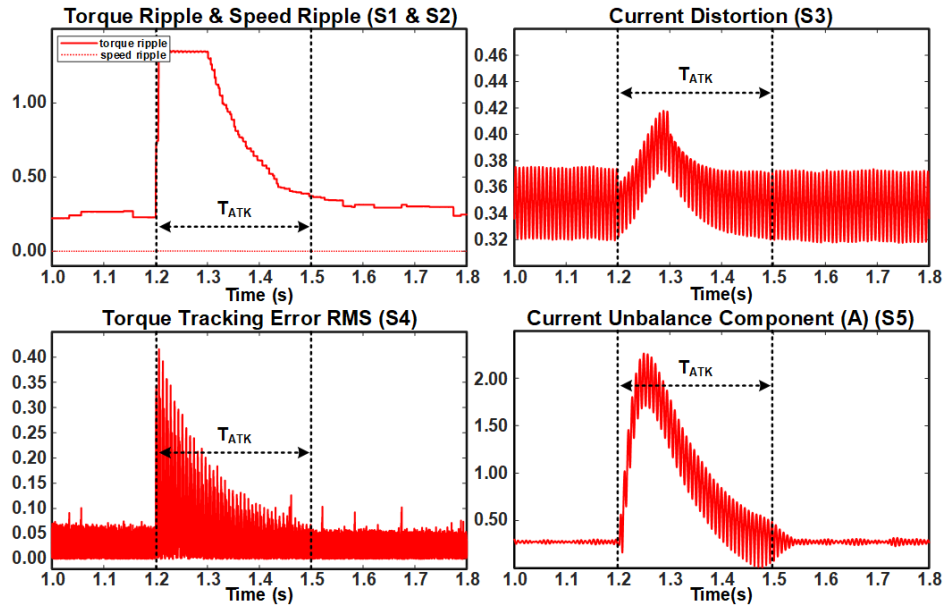


Figure 2.8: Case 4: decaying high frequency harmonics, $\hat{y} = y + 25e^{-t/0.1} \cdot \sin(2\pi \cdot 200 \cdot t), t \in \boldsymbol{T_{ATK}}$, targeting phase A.

Case 5: $\hat{y} = y + \mathscr{F}(t), t \in \boldsymbol{T_{ATK}}$, targeting phase A

Case 5 discusses the attack with a periodic pulse signal defined by Eq.(2.12), where $f = 1000Hz, K = 30A, D = 0.25$. As the results shown in Fig.2.9, the system will have a periodic fluctuation with the similar frequency of the attack. Like case 4, this feature could be used to determine if the attacks are periodic. From case 4 and case 5, it is obvious that when some false signals are injected to the sensor signals, the metrics response will have similar pattern to the injected signals. This could be used as one of the detection and diagnosis criteria.

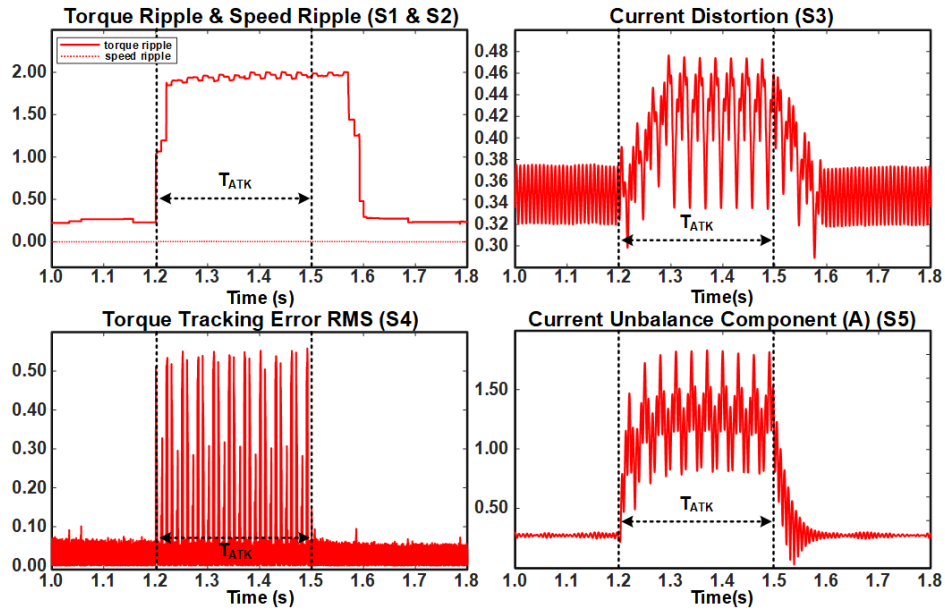


Figure 2.9: Case 5: periodic pulse injection, $\hat{y} = y + \mathscr{F}(t), t \in \boldsymbol{T_{ATK}}$, targeting phase A.

Case 8: $\hat{y} = y +$ white noise, $t \in \boldsymbol{T_{ATK}}$, targeting phase A and phase B

Case 8 talks about double phases attack with white noise. Such attacks are rather more difficult to detect and diagnosis, because white noise is a inherent attribute for all sensors. It could easily trick people into wrong diagnostic conclusions, such as device aging or environment change. The results are shown in Fig.2.10. When the same noise is injected to two phases (phase A and phase B), the three phase current balance is damaged, but the current distortion is likely to maintain the healthy conditions as the noise power is relatively small. However, such small power noise is able to generate significant torque ripple. This case shows that not all evaluation metrics could reflect a deliberate designed attack, so the detection and diagnosis process need to consider enough evaluation metrics to come up with the accurate conclusion.

Case 13: $\hat{y} = y +$ white noise, $t \in \boldsymbol{T_{ATK}}$, targeting phase A, phase B and phase C

When the noise is injected into all three sensors, the results of case 13 are shown in Fig.2.11. It is hard to distinguish the attack duration from the metrics waveform as the white noise power is relatively small. So this kind of attacks may not change the system operating conditions a lot. However, this also means this kind of attacks are hard to detect. In this case, the white noise could be brought by cyber attacks like

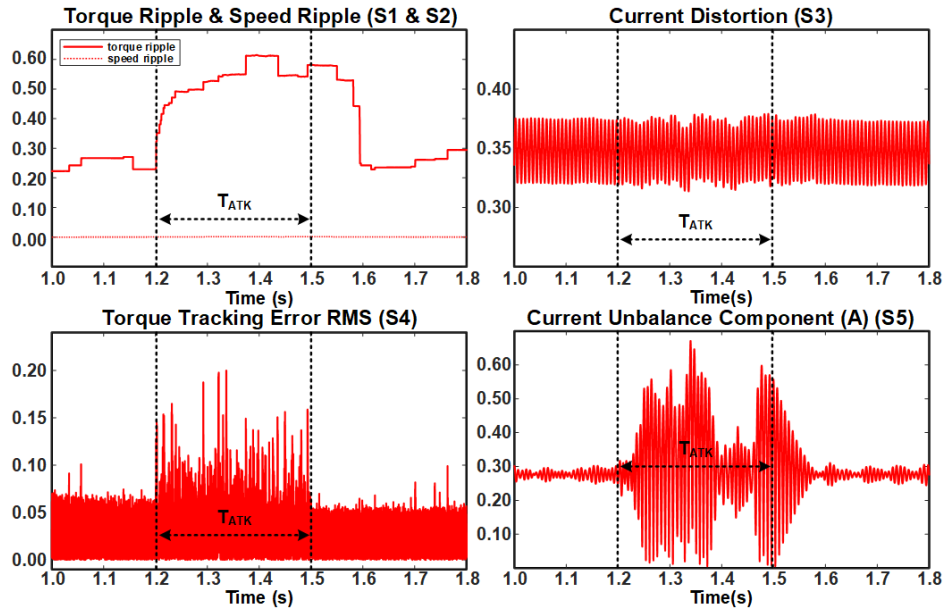


Figure 2.10: Case 8: $\hat{y} = y +$ white noise, $t \in \boldsymbol{T_{ATK}}$, targeting phase A and phase B.

interception, where the system operation is not affected but the system information is lost. Meanwhile, once the noise power becomes larger, it also could make the system unstable as demonstrated in previous sections.

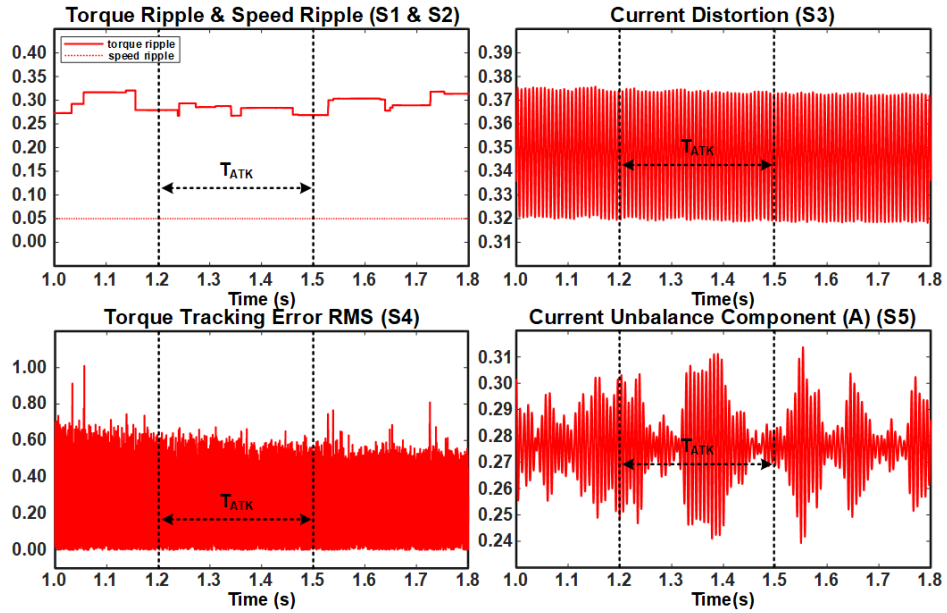


Figure 2.11: Case 13: $\hat{y} = y +$ white noise, $t \in \boldsymbol{T_{ATK}}$, targeting phase A, B and C.

From the features of simulation results and the impact analysis of several typical cyber attacks, an guideline for the sensor attack detection and diagnosis can be summarized:

1. When one or more of the aforementioned performance metrics have drastic variation, there is likely an attack targeting on the system.
2. If the metrics profiles maintain similar to the healthy operation conditions, it could be assumed that the system is not under attacks or the attack is minor and do not harm the system operating performance.
3. When profile of the current distortion S_3 has two obvious spikes, there is likely an attack defined by Eq.(2.8). Then its variation pattern could be used to determine if the attack increases or reduces the feedback signals.
4. If the current unbalance component (S_5) profile has a huge jump, it is likely a single phase attack.
5. If the current unbalance component (S_5) is small or does not obviously change and other metrics ($S_1 - S_4$) have abnormal profiles, it is likely a three phase attack.
6. If the metrics have a decaying feature, the attack is likely to have the same decaying signal;
7. If the metrics have a periodic feature, the attack is likely an periodic signal with the same frequency.

Table 2.3: Simulation Results of ATK I

α	0.5	0.6	0.7	0.8	0.9
$\mathcal{K}_{Imp}$	88.929	65.924	46.015	27.397	11.319
α	1.1	1.2	1.3	1.4	1.5
$\mathcal{K}_{Imp}$	10.359	24.145	36.274	48.755	61.776

Table 2.4: Simulation Results of ATK II

β	-30	-20	-10	-5
$\mathcal{K}_{Imp}$	82.839	52.774	24.773	11.090
β	5	10	20	30
$\mathcal{K}_{Imp}$	10.931	24.460	53.273	84.070

Vulnerability Assessments of Different Attacks

In order to comprehensively assess the system vulnerability due to sensor data integrity attacks through the evaluation metrics and impact index we proposed, two types of common attacks modeled by Eq. 2.8 (ATK I) and Eq. 2.9 (ATK II) are simulated with $\alpha = 0.5 \cdot \cdot 1.5$ and $\beta = -30 \cdot \cdot 30$. The simulation results are shown in Table 2.3 and Table 2.4, respectively.

From the results shown in Table 2.3, Table 2.4 and Fig. 2.12, a ground truth could be proved that the more deviation an attack could cause, the more sever impact it will bring to the systems. It should be noticed that the case where $\alpha < 0$ is not taking into consideration, because in such case, the feedback control will become positive, which means the system will be unstable, and then such attacks could be easily dealt with by protection components like relays.

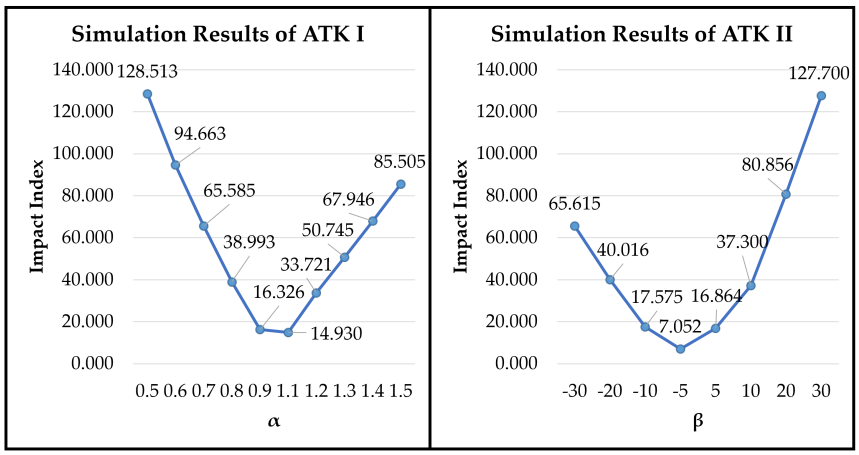


Figure 2.12: Statistical diagram of table III and table IV.

Meanwhile, a statistical graph based on the results in Table 2.2 is shown in Fig. 2.13. In this diagram, each impact index is analyzed independently in each cases. Then, we can make the following conclusions:

1. Three phase attacks will barely have impact on the electric drive system as three phase bias are filtered by DQZ transformation. Nevertheless, these kinds of attacks may also cause security issues from other point of view, such as information stealing.

2. The impact of white noise attacks is relatively smaller, which means such cyber attacks are more difficult to detect. Besides, the impact of white noise is also dependent on the noise energy.

3. Except for three phase additional attacks like case 13-15, multiple phase attacks could casue more sever impact to the systems.

4. Among 15 cases we proposed, none of them has a drastic impact on $k_{imp}^{S_3}$, which means that these attacks will not drastically increase the current distortion. Thus, we could come to the conclusion that increasing current harmonics distortion requires more sophisticating attacks.

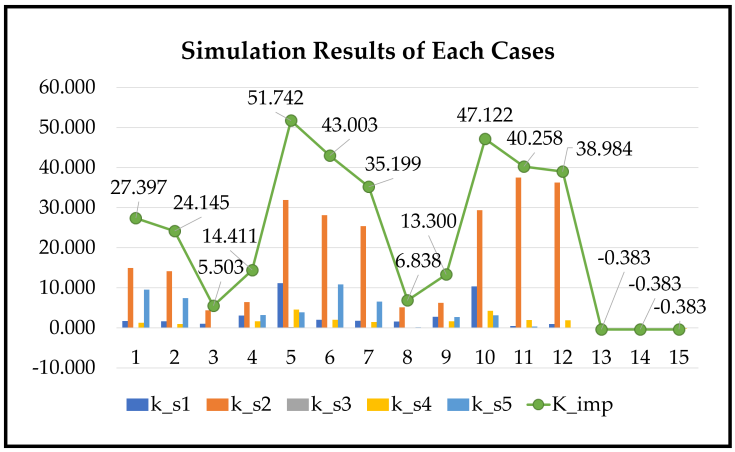


Figure 2.13: Statistical graph of the simulation results.

2.2 Mathematical Models for Dynamic Systems

After acquiring an empirical impression of the cyberattack impacts on intelligent electric drive systems, the next step will be a systematic analysis of the underlying mechanisms of various cyberattacks and the differences and similarities of existing research on traditional physical fault analysis. Essentially, electric drive systems are dynamic systems achieving energy conversions between electrical and mechanical forms accompanied by dissipation in the form of heat. Additionally, such processes are monitored, controlled, and optimized by adding intelligent algorithms enabled by sensors and digital control units. As such, engineers and scientists require some common frameworks for describing and analyzing these dynamic systems. These frameworks are referred to as mathematical models, which are abstracts of the target electric drive systems. Therefore, the model precision is subject to the assumptions and requirements made by the engineers and scientists. As suggested by Albert Einstein, "Everything should be made as simple as possible, but not simpler." That is, the mathematical models should be simple but sufficiently detailed to suit specific research questions.

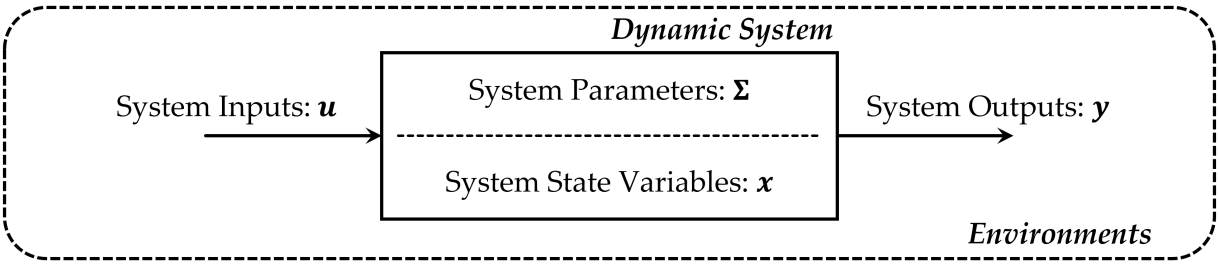


Figure 2.14: General model for dynamic systems.

Figure 2.14 shows a diagram of generic dynamic systems. The dynamic system is characterized by a set of state variables $\boldsymbol{x}$ associated with system parameters $\boldsymbol{\Sigma}$. The state variables are influenced by system inputs $\boldsymbol{u}$, which represent the controlled and uncontrolled actions from the external environments. The system outputs $\boldsymbol{y}$ represent the observable or measurable behaviors of the dynamic systems. Currently, there are five research targets corresponding to different known variables.

1. **Simulation and Prediction.** Given a future trajectory of inputs $\boldsymbol{u}$, initial state variables $\boldsymbol{x}_0$, and the system parameters $\boldsymbol{\Sigma}$, then predict the output trajectories $\boldsymbol{y}$.

2. **State Estimation.** Given history trajectories of inputs $\boldsymbol{u}$ and outputs $\boldsymbol{y}$ associated with the system parameters $\boldsymbol{\Sigma}$, find state trajectories that are consistent with $\boldsymbol{u}$, $\boldsymbol{y}$, and $\boldsymbol{\Sigma}$.

3. **Design and Planning.** Given the input information $\boldsymbol{u}$ and desired output responses $\boldsymbol{y}$, find system parameters $\boldsymbol{\Sigma}$ that $\boldsymbol{u}$ acting on $\boldsymbol{\Sigma}$ will produce $\boldsymbol{y}$.

4. **Model Identification.** Given the measured inputs $\boldsymbol{u}$ and outputs $\boldsymbol{y}$, determine the model parameters $\boldsymbol{\Sigma}$.

5. **Control Synthesis.** Given system parameters $\boldsymbol{\Sigma}$, initial states $\boldsymbol{x}_0$, and desired output responses $\boldsymbol{y}$, find suitable inputs $\boldsymbol{u}$ that will produce the desired outputs $\boldsymbol{y}$.

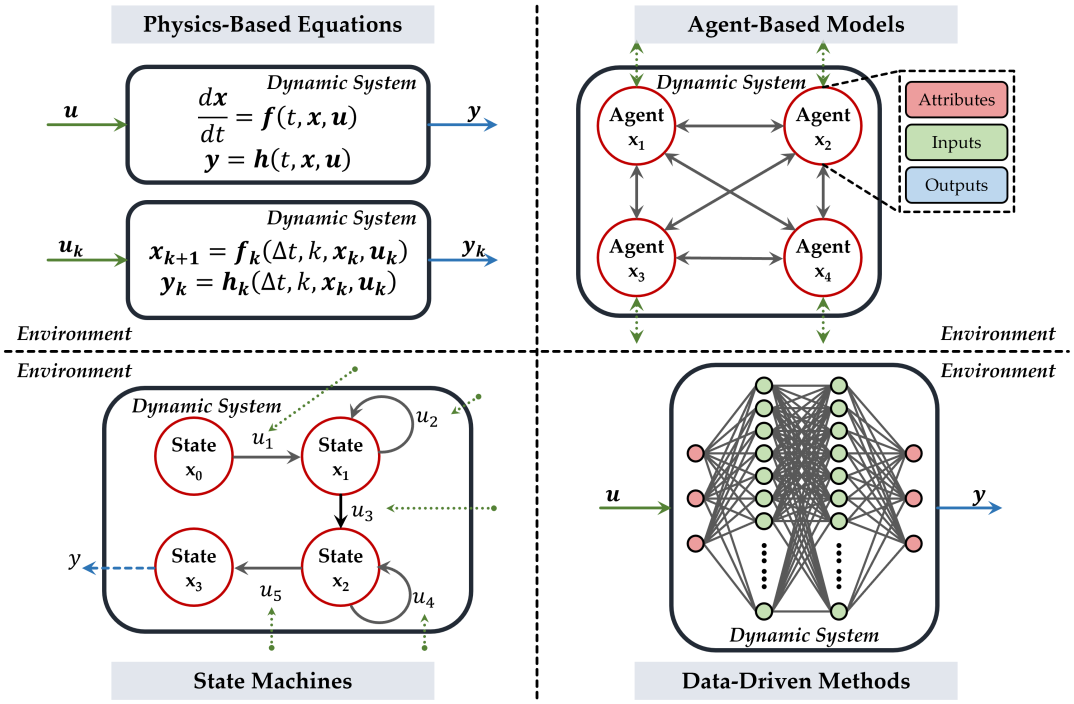


Figure 2.15: Diagrams of the four common modeling paradigms.

With the increasing complexity of modern dynamic systems, the modeling approaches also evolved in the past decades to incorporate more desired details, such as discrete dynamics and nonlinearities. So far, there are four primary modeling approaches: physics-based equations, state machines, agent-based models, and data-driven methods. (Rai & Sahu, 2020) Physics-based equations and state machines are used to model a system's continuous and discrete dynamics, respectively. They are purely based on the physical dynamics of the system. Agent-based models leverage the data generated by the system, along with the knowledge of target systems. The data-driven methods depend purely on the acquired data sets. The following subsections include a brief critical appraisal of these modeling paradigms, and Figure 2.15 shows a conceptual diagram of these paradigms.

2.2.1 Physics-Based Equations

The physics-based equations were initially designed to model continuous physical dynamics evolving with time. These equations can be simple dependence equations, ordinary differential equations, partial differential equations, or a combination of them. Causality, conservation laws, objectivity, and composability are the main attributes of physics-based equations. For electric drive systems, the state-space model is one of the most widely adopted physics-based equations. For a general time(t) dependent system with n_x state variables $\boldsymbol{x} \in \mathbb{R}^{n_x}$, n_u input variables $\boldsymbol{u} \in \mathbb{R}^{n_u}$, and n_y output variables $\boldsymbol{y} \in \mathbb{R}^{n_y}$, the state-space model takes the form of:

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}) \tag{2.22a}$$

$$\boldsymbol{y} = \boldsymbol{h}(t, \boldsymbol{x}, \boldsymbol{u}) \tag{2.22b}$$

where $\boldsymbol{f}$ is the state equation and $\boldsymbol{h}$ is the output equation.

Initially, these equation-based models are limited to modeling continuous dynamics. However, to incorporate the development of digital controllers, which are inherently discrete, multiple discretization methods are proposed, such as Euler methods and Runge-Kutta methods. The discretized state-space model takes the form of:

$$\boldsymbol{x}(k+1) = \boldsymbol{f}_d(\Delta t, k, \boldsymbol{x}(k), \boldsymbol{u}(k)) \tag{2.23a}$$

$$\boldsymbol{y}(k+1) = \boldsymbol{h}_d(\Delta t, k, \boldsymbol{x}(k), \boldsymbol{u}(k)) \tag{2.23b}$$

where Δt is the step size in the time scale, $\boldsymbol{f}_d$ and $\boldsymbol{h}_d$ are the discretized state and output equations.

The discrete state-space model is an approximation of the continuous dynamics. Therefore, the accuracy of discrete models depends on discretization methods and time steps. Meanwhile, stability analysis for continuous models and discrete models are different. The stability of the discrete system implies that the corresponding continuous system is also stable. However, the stability of the continuous system cannot imply the stability of the corresponding discrete system for all time steps Δt. (Bahill & Szidarovszky, 2009) As the discretized physics-based equations provide convenience for incorporating digital processes, they are widely adopted in control law designs for electric drive systems. At the same

time, continuous models are frequently used in analysis related to physical plants, such as various fault mechanisms.

2.2.2 State Machines

Unlike the discretized state-space model, which approximates the corresponding continuous dynamics, state machines are designed to model those inherently discrete dynamics, such as digital controllers and logic circuits. (Bahill & Szidarovszky, 2009) One of the most widely used state machine models is finite state machines, which model discrete dynamic systems with a limited number of states. (Rai & Sahu, 2020) Mathematically, a finite state machine is a quintuple $(\Sigma, \Gamma, S, s_0, \delta, \omega)$, where:

- Σ is the input alphabet (a finite non-empty set of symbols)
- Γ is the output alphabet (a finite non-empty set of symbols)
- S is a finite, non-empty set of states
- s_0 is one or a set of initial sets which is an element or a subset of S
- δ is the state transition function: $\delta : S \times \Sigma \rightarrow S$
- ω is the output function

Finite state machines could be categorized into Mealy Machines ($\omega : \S \times \Sigma \rightarrow \Gamma$) and Moore Machines ($\omega : \S \rightarrow \Gamma$) depending on whether or not outputs are related to inputs.

While most studies of electric drive systems adopted continuous and discrete state-space models, some research is leveraging finite state machines to develop control and monitoring solutions for electric drives, such as predictive torque control, adaptive control, and IGBT transient modeling.

2.2.3 Agent-Based Models

With the recent development of perception, communication, and computation technologies, distributed cooperative control among multiple dynamic systems is getting more attention. (D. Zhang et al., 2021) Applications like intelligent manufacturing and transportation systems require multiple electric drive systems connecting and sharing information within interactive networks. Agent-based models represent such connected dynamic systems using a set of agents and semantically defined rules. Systems modeled by such methods are also referred to as multi-agent systems (MAS). Generally, the agent is defined as a piece of software code with intelligent decision-making functionalities. An agent can be considered a self-directed object capable of autonomously choosing actions based on its situation. The construction of an agent-based model can be broken down into the following steps. (Elçi et al., 2011; Leitao et al., 2016)

1. Defining the agents in the context of the system; identifying attributes of the agent; and other classes, along with their attributes.

2. Defining the environment where the agents reside and the objects with which the agents interact.

3. Designing the methods by which agent attributes will be updated in response to agent-to-agent interactions or agent interaction with the environment.

4. Implementing the designed agent model in modeling software.

While agent-based models promise to capture the intra- and inter-dependencies among cyber and physical components and the heterogeneity and complexity of the connected dynamic systems, the physical dynamics are usually highly abstract. Therefore, agent-based models have yet to be applied to electric drive systems. In most MASs, electric drives are still being treated as actuators serving the larger cyber-physical system.

2.2.4 Data-Driven Methods

Data-driven methods explore system dynamics purely with observed and collected data. Such methods are categorized into supervised approaches (using both system input and output data) and unsupervised approaches (using only system output data). The core of most data-driven methods is to explore relations, structures, and patterns from available data sets with suitable architectures and optimization algorithms. Standard models include but are not limited to autoregressive integrated moving average models (ARIMA), support vector machines (SVM), random forests (RF), clustering, and deep neural networks (DNN). (Giraldo et al., 2018) However, most industrial applications are still conservative about implementing data-driven approaches to mission-critical systems. The reasons could be summarized as the followings.

1. Industrial systems work in dynamic environments where system behaviors changes continuously.

2. The data-driven methods cannot be generalized beyond their initial set of training data.

3. There are limited available data sets in most industrial applications.

4. Data-driven methods are agnostic to underlying physics, resulting in predictions inconsistent with the physics law.

Luckily, recent advancements in physics-guided machine learning methods shed light on the future. These hybrid methods incorporate existing physics information into the machine learning architectures to make the original black-box model a grey-box model. For electric drive systems, data-driven models are mostly related to fault dynamics, such as pattern recognition methods. (Karniadakis et al., 2021) However, with the progress in physics-guided machine learning, data-driven methods are raising interest in other applications, such as advanced nonlinear controls.

The four standard modeling paradigms for dynamics shown in Figure 2.15 have their respective focus. From the perspective of system safety and security of electric drive systems, the target is to analyze the failure mechanisms and explore the underlying patterns of different anomalies. Physical fault analysis

for electric drive systems is the early research topic targeting system safety. Various mathematical models targeting different types of fault mechanisms were proposed. Most of such models were derived from physics-based equations. Therefore, this chapter will first explore typical fault models from physics-based equations and then extend such models to incorporate recently raised cybersecurity concerns. However, as cyber security issues originate from the digital domain, more than physics-based equations are needed, and assistance is desired from other modeling approaches. The result will form a hybrid model, which will be discussed following the physics-based equation models. In the end, impact analysis with comprehensive case studies is carried out to provide some empirical insights to complement the theoretical mathematical models.

2.3 Physics-Based Equations for Common Physical Faults in Electric Drive Systems

Analysis of electric drive faults and failures could date back decades ago. Most of this analysis originated from the physics-based equations for underlying mechanisms.(Toliyat et al., 2012) Therefore, equation-based models will be a great starting point for analyzing electric drive systems' safety and security problems. However, it is not wise to dive into all types of physical fault cases since electric drive systems have hundreds of, if not thousands, different physical fault scenarios. In addition, the overall goal is to analyze the differences and impacts brought by recently emerged cyber-security issues. Therefore, this section will focus on two of the most widely appeared physical faults in electric drive systems: inter-turn short circuit faults (ITSC) and mechanical bearing faults (MBF). The purpose is to provide an analytical basis for existing safety and security studies for electric drive systems.

2.3.1 Fault Model I: Motor Winding Inter-Turn Short Circuit Fault in Pamanent Magnet Synchronous Machines

This section will first derive the equation-based model for ITSC in a permanent magnet synchronous machine (PMSM) from its equivalent circuit. Then, the results will be formalized to a general state-space model.(Romeral et al., 2011)

Machine Model in Healthy Conditions

Before digging into the fault model, the PMSM machine model in healthy conditions is established first.

$$\boldsymbol{v}_{s,abc} = \boldsymbol{R}_s \cdot \boldsymbol{i}_{s,abc} + \frac{d}{dt}\boldsymbol{\lambda}_{s,abc} \tag{2.24}$$

$$\boldsymbol{\lambda}_{s,abc} = \boldsymbol{L}_s \cdot \boldsymbol{i}_{s,abc} + \boldsymbol{\lambda}_{PM,abc} \tag{2.25}$$

where:
$\boldsymbol{v}_{s,abc} = (v_a, v_b, v_c)^T$ and $\boldsymbol{i}_{s,abc} = (i_a, i_b, i_c)^T$ are voltage and current vectors.

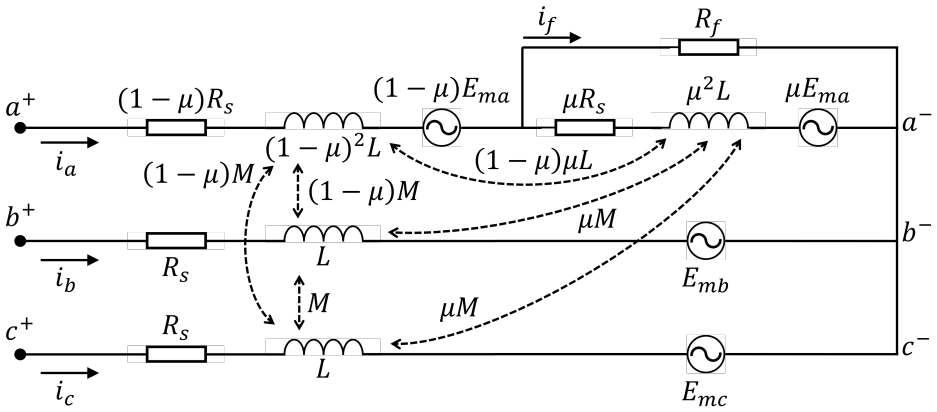


Figure 2.16: The equivalent circuit of the ITSC in a PMSM.

$\boldsymbol{L}_s = \begin{pmatrix} L_{aa} & M_{ab} & M_{ac} \\ M_{ba} & L_{bb} & M_{bc} \\ M_{ca} & M_{cb} & L_{cc} \end{pmatrix}$ is the inductance matrix, and the self- and mutual- inductances are calculated from:

$$L_{aa} = L_{le} + L_{g1} + L_{g2} \cdot \cos(2\theta)$$
$$L_{bb} = L_{le} + L_{g1} + L_{g2} \cdot \cos(2\theta - \frac{2\pi}{3})$$
$$L_{cc} = L_{le} + L_{g1} + L_{g2} \cdot \cos(2\theta + \frac{2\pi}{3})$$
$$M_{ab} = M_{ba} = -\frac{1}{2}L_{g1} + L_{g2} \cdot \cos(2\theta - \frac{2\pi}{3})$$
$$M_{bc} = M_{cb} = -\frac{1}{2}L_{g1} + L_{g2} \cdot \cos(2\theta)$$
$$M_{ca} = M_{ac} = -\frac{1}{2}L_{g1} + L_{g2} \cdot \cos(2\theta + \frac{2\pi}{3})$$

where L_{le} is the leakage inductance, L_{g1} and L_{g2} are the constant and variable part of the air-gap inductance.

$\boldsymbol{R}_s = \begin{pmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{pmatrix}$ is the resistance matrix where R_s is the winding resistance.

$\boldsymbol{\lambda}_{PM,abc} = \lambda_{PM} \cdot \begin{pmatrix} \cos(\theta) \\ \cos(\theta - \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) \end{pmatrix}$ is the permanent magnet flux vecotor where λ_{PM} is the permanent magnet flux linkage.

And θ is the synchronous electrical angular position.

For simplification, a common approach is to transfer the above equations in ABC reference frame to the DQZ reference frame, namely DQZ transformation. The transformation matrix is defined as follows:

$$\boldsymbol{T} = \begin{pmatrix} \frac{2}{3}\cos(\theta) & \frac{2}{3}\cos(\theta - \frac{2\pi}{3}) & \frac{2}{3}\cos(\theta + \frac{2\pi}{3}) \\ -\frac{2}{3}\sin(\theta) & -\frac{2}{3}\sin(\theta - \frac{2\pi}{3}) & -\frac{2}{3}\sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \tag{2.26}$$

so that $\boldsymbol{T} \cdot (S_a, S_b, S_c)^T = (S_d, S_q, S_0)^T$. Thus, the inverse transform matrix is:

$$\boldsymbol{T}^{-1} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 1 \\ \cos(\theta - \frac{2\pi}{3}) & -\sin(\theta - \frac{2\pi}{3}) & 1 \\ \cos(\theta + \frac{2\pi}{3}) & -\sin(\theta + \frac{2\pi}{3}) & 1 \end{pmatrix} \tag{2.27}$$

Therefore, after transformation, system Eq. (2.24) and Eq. (2.25) become:

$$\boldsymbol{v}_{s,DQ} = \boldsymbol{R}_{s,DQ} \cdot \boldsymbol{i}_{s,DQ} + \frac{d}{dt}\boldsymbol{\lambda}_{s,DQ} + \boldsymbol{A}_c \cdot \boldsymbol{\lambda}_{s,DQ} \tag{2.28}$$

$$\boldsymbol{\lambda}_{s,DQ} = \boldsymbol{L}_{s,DQ} \cdot \boldsymbol{i}_{s,DQ} + \boldsymbol{\lambda}_{PM,DQ} \tag{2.29}$$

where:
$\boldsymbol{v}_{s,DQ} = (v_d, v_q, v_0)^T$ and $\boldsymbol{i}_{s,DQ} = (i_d, i_q, i_0)^T$ are voltage and current vectors.
$\boldsymbol{L}_{s,DQ} = \begin{pmatrix} L_d & 0 & 0 \\ 0 & L_q & 0 \\ 0 & 0 & L_0 \end{pmatrix}$ is the inductance matrix, and the inductances are calculated from:

$$L_d = L_{le} + \frac{3}{2}(L_{g1} + L_{g2})$$
$$L_q = L_{le} + \frac{3}{2}(L_{g1} - L_{g2})$$
$$L_0 = L_{le}$$

$\boldsymbol{R}_{s,DQ} = \begin{pmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{pmatrix}$ is the resistance matrix.
$\boldsymbol{\lambda}_{PM,DQ} = \lambda_{PM} \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ is the permanent magnet flux vecotor.
$\boldsymbol{A}_c = \begin{pmatrix} 0 & -\omega & 0 \\ \omega & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ is the flux coupling matrix where ω is the electrical synchronous speed in rad/s.

Machine Model in Fault Conditions

When the fault is appearing, one extra dimension is introduced to the system. In order to seperate the fault dynamics and the original system dynamics, we will first establish the overall dynamics in the ABC reference frame. According to the euqivalent circuit in Fig. 2.16, the system equations under ITSC fault could be written as followings:

$$v_{a,f} = R_s \cdot i_{a,f} - \mu R_s \cdot i_f + \frac{d}{dt}\lambda_{a,f} \tag{2.30}$$

$$v_{b,f} = R_s \cdot i_{b,f} + \frac{d}{dt}\lambda_{a,f} \tag{2.31}$$

$$v_{c,f} = R_s \cdot i_{b,f} + \frac{d}{dt}\lambda_{a,f} \tag{2.32}$$

$$\lambda_{a,f} = L_{aa} \cdot i_{a,f} + M_{ab} \cdot i_{b,f} + M_{ac} \cdot i_{c,f} - \mu L_{aa} \cdot i_f + \lambda_{PM} \cdot \cos(\theta) \tag{2.33}$$

$$\lambda_{b,f} = M_{ba} \cdot i_{a,f} + L_{bb} \cdot i_{b,f} + M_{bc} \cdot i_{c,f} - \mu M_{ba} \cdot i_f + \lambda_{PM} \cdot \cos(\theta - \frac{2\pi}{3}) \tag{2.34}$$

$$\lambda_{c,f} = M_{ca} \cdot i_{a,f} + M_{cb} \cdot i_{b,f} + L_{cc} \cdot i_{c,f} - \mu M_{ca} \cdot i_f + \lambda_{PM} \cdot \cos(\theta + \frac{2\pi}{3}) \tag{2.35}$$

$$v_f = 0 = -(R_f + \mu R_s) \cdot i_f + \mu R_s \cdot i_{a,f} + \frac{d}{dt}\lambda_f \tag{2.36}$$

$$\lambda_f = \mu L_{aa} \cdot i_{a,f} + \mu M_{ab} \cdot i_{b,f} + \mu M_{ac} \cdot i_{c,f} - \mu^2 L_{aa} \cdot i_f + \mu \lambda_{PM} \cdot \cos(\theta) \tag{2.37}$$

After stacking Eq. (2.30) - Eq. (2.37), we could get the matrix form of the system dynamics.

$$\boldsymbol{v}_{sf,abc} = \boldsymbol{R}_s \cdot \boldsymbol{i}_{sf,abc} + \frac{d}{dt}\boldsymbol{\lambda}_{sf,abc} - \mu \cdot \boldsymbol{R}_s \cdot \boldsymbol{A}_{f1} \cdot i_f \tag{2.38}$$

$$\boldsymbol{\lambda}_{sf,abc} = \boldsymbol{L}_s \cdot \boldsymbol{i}_{sf,abc} + \boldsymbol{\lambda}_{PM,abc} - \mu \cdot \boldsymbol{L}_s \cdot \boldsymbol{A}_{f1} \cdot i_f \tag{2.39}$$

$$v_f = 0 = -(R_f + \mu R_s) \cdot i_f + \frac{d}{dt}\lambda_f + \mu \cdot \boldsymbol{A}_{f1}^T \cdot \boldsymbol{R}_s \cdot \boldsymbol{i}_{sf,abc} \tag{2.40}$$

$$\lambda_f = -\mu^2 L_{aa} \cdot i_f + \mu \cdot \boldsymbol{A}_{f1}^T \cdot \boldsymbol{L}_s \cdot \boldsymbol{i}_{sf,abc} + \mu \cdot \boldsymbol{A}_{f1}^T \cdot \boldsymbol{\lambda}_{PM,abc} \tag{2.41}$$

where $\boldsymbol{A}_{f1} = (1, 0, 0)^T$.

Then, re-do the DQZ transformation to Eq. (2.38) - Eq. (2.41). We could get:

$$\boldsymbol{v}_{sf,DQ} = \boldsymbol{R}_{s,DQ} \cdot \boldsymbol{i}_{sf,DQ} + \frac{d}{dt}\boldsymbol{\lambda}_{sf,DQ} + \boldsymbol{A}_c \cdot \boldsymbol{\lambda}_{sf,DQ} - \mu \cdot \boldsymbol{R}_{s,DQ} \cdot \boldsymbol{T}_{f1} \cdot i_f \tag{2.42}$$

$$\boldsymbol{\lambda}_{s,DQ} = \boldsymbol{L}_{s,DQ} \cdot \boldsymbol{i}_{s,DQ} + \boldsymbol{\lambda}_{PM,DQ} - \mu \cdot \boldsymbol{L}_{s,DQ} \cdot \boldsymbol{T}_{f1} \cdot i_f \tag{2.43}$$

$$v_f = 0 = -(R_f + \mu R_s) \cdot i_f + \frac{d}{dt}\lambda_f + \mu \cdot \boldsymbol{T}_{f2} \cdot \boldsymbol{R}_{s,DQ} \cdot \boldsymbol{i}_{sf,DQ} \tag{2.44}$$

$$\lambda_f = -\mu^2 L_{aa} \cdot i_f + \mu \cdot \boldsymbol{T}_{f2} \cdot \boldsymbol{L}_{s,DQ} \cdot \boldsymbol{i}_{sf,DQ} + \mu \cdot \boldsymbol{T}_{f2} \cdot \boldsymbol{\lambda}_{PM,DQ} \tag{2.45}$$

where $\boldsymbol{T}_{f1} = \boldsymbol{T} \cdot \boldsymbol{A}_{f1}$ and $\boldsymbol{T}_{f2} = \boldsymbol{A}_{f1}^T \cdot \boldsymbol{T}^{-1}$.

State-Space Model of ITSC Faults in PMSM

In order to generalize the system dynamic model described in Eq. (2.42) - Eq. (2.45), we proposed a state-space model for ITSC faults in PMSM. In the proposed model, the original system dynamics in healthy conditions are seperated from the fault dynamics. Then, these two dynamics form two different state-space models, which are linked through the input and output variables. These two models are named as 'Basic Model' and 'Fault Model'. For both models, the state variables are the flux linkages, i.e. $\boldsymbol{\lambda}_{s,DQ}$ and λ_f. The input variables of Basic Model are the motor input voltages $\boldsymbol{v}_{sf,DQ}$ and the fault current i_f. The output variables of Basic Model are the motor currents $\boldsymbol{i}_{sf,DQ}$. Meanwhile, the input variables of Fault Model are the motor currents $\boldsymbol{i}_{sf,DQ}$, and the output variable is the fault current i_f. Based on these variables, Eq. (2.42) - Eq. (2.45) are re-written as:

$$\text{Basic} \begin{cases} \frac{d}{dt}\boldsymbol{x}_1 = \boldsymbol{A}_{B1} \cdot \boldsymbol{x}_1 + \boldsymbol{B}_{B1} \cdot \boldsymbol{u}_1 + \boldsymbol{G}_{B1} \cdot \boldsymbol{w}_1 \\ \boldsymbol{y}_1 = \boldsymbol{C}_{B1} \cdot \boldsymbol{x}_1 + \boldsymbol{F}_{B1} \cdot \boldsymbol{f}_1 + \boldsymbol{H}_{B1} \cdot \boldsymbol{w}_1 \end{cases} \tag{2.46}$$

$$\text{Fault} \begin{cases} \frac{d}{dt}\boldsymbol{s}_1 = \boldsymbol{A}_{F1} \cdot \boldsymbol{s}_1 + \boldsymbol{B}_{F1} \cdot \boldsymbol{y}_1 + \boldsymbol{G}_{F1} \cdot \boldsymbol{w}_1 \\ \boldsymbol{f}_1 = \boldsymbol{C}_{F1} \cdot \boldsymbol{s}_1 + \boldsymbol{D}_{F1} \cdot \boldsymbol{y}_1 + \boldsymbol{H}_{F1} \cdot \boldsymbol{w}_1 \end{cases} \tag{2.47}$$

where $\boldsymbol{x}_1 = \boldsymbol{\lambda}_{s,DQ}$, $\boldsymbol{y}_1 = \boldsymbol{i}_{sf,DQ}$, $\boldsymbol{u}_1 = \boldsymbol{v}_{sf,DQ}$, $\boldsymbol{f}_1 = i_f$, $\boldsymbol{w}_1 = \boldsymbol{\lambda}_{PM,DQ}$, and $\boldsymbol{s}_1 = \lambda_f$. And the parameter matrices are $\boldsymbol{A}_{B1} = -(\boldsymbol{R}_{s,DQ} \cdot \boldsymbol{L}_{s,DQ}^{-1} + \boldsymbol{A}_c)$, $\boldsymbol{B}_{B1} = \boldsymbol{I}$, $\boldsymbol{G}_{B1} = \boldsymbol{R}_{s,DQ} \cdot \boldsymbol{L}_{s,DQ}^{-1}$, $\boldsymbol{C}_{B1} = \boldsymbol{L}_{s,DQ}^{-1}$, $\boldsymbol{F}_{B1} = \mu \boldsymbol{T}_{f1}$, $\boldsymbol{H}_{B1} = -\boldsymbol{L}_{s,DQ}^{-1}$,$\boldsymbol{A}_{F1} = (R_f + \mu R_s)(\mu^2 L_{aa})^{-1}$, $\boldsymbol{B}_{F1} = -\boldsymbol{T}_{f2}((R_f + \mu R_s)(\mu^2 L_{aa})^{-1} \cdot \boldsymbol{L}_{s,DQ} + \mu \boldsymbol{R}_{s,DQ})$, $\boldsymbol{G}_{F1} = (R_f + \mu R_s)(\mu^2 L_{aa})^{-1} \cdot \boldsymbol{T}_{f2}$, $\boldsymbol{C}_{F1} = -(\mu^2 L_{aa})^{-1}$, $\boldsymbol{D}_{F1} = (\mu^2 L_{aa})^{-1} \cdot \boldsymbol{T}_{f2} \cdot \boldsymbol{L}_{s,DQ}$, and $\boldsymbol{H}_{F1} = (\mu^2 L_{aa})^{-1} \cdot \boldsymbol{T}_{f2}$.

2.3.2 Fault Model II: Motor Bearing Fault (MBF) in Induction Machines

Motor bearing faults (MBF) are one of the primary fault conditions appearing in the induction machine. Such faults account for more than 50% of the total failure cases in real-world applications. Bearing faults are usually casued by mechanical frictions and chemical corrosion. When a bearing fault appears, it will cause the eccentricity to the rotor shaft and break the symmetricy among all the phase windings. Such eccentricity will cause the inductance in each phase varying with respect to the rotor position. Details will be provided later in this section.

Machine model in healthy conditions

As the DQZ transformation for IM is similar to the one in PMSM, this section will directly assume all the variables have been transferred to DQZ reference frame. Then, the system equations of IM under healthy

conditions is expressed as followings: (S. Zhang et al., 2020)

$$v_{ds} = R_s i_{ds} + \frac{d}{dt}\lambda_{ds} - \omega_s \lambda_{qs} \tag{2.48}$$

$$v_{qs} = R_s i_{qs} + \frac{d}{dt}\lambda_{qs} + \omega_s \lambda_{ds} \tag{2.49}$$

$$0 = R_r i_{dr} + \frac{d}{dt}\lambda_{dr} - (\omega_s - \omega_r)\lambda_{qr} \tag{2.50}$$

$$0 = R_r i_{qr} + \frac{d}{dt}\lambda_{qr} + (\omega_s - \omega_r)\lambda_{dr} \tag{2.51}$$

$$\lambda_{ds} = L_s i_{ds} + L_m i_{dr} \tag{2.52}$$

$$\lambda_{qs} = L_s i_{qs} + L_m i_{qr} \tag{2.53}$$

$$\lambda_{dr} = L_m i_{ds} + L_r i_{dr} \tag{2.54}$$

$$\lambda_{qr} = L_m i_{qs} + L_r i_{qr} \tag{2.55}$$

Stack Eq. (2.48) to Eq. (2.55) in matrix forms.

$$\boldsymbol{v}_{DQ} = \boldsymbol{R}_{DQ} \cdot \boldsymbol{i}_{DQ} + \frac{d}{dt}\boldsymbol{\lambda}_{DQ} + \boldsymbol{A}_{c1} \cdot \boldsymbol{\lambda}_{DQ} \tag{2.56}$$

$$\boldsymbol{\lambda}_{DQ} = \boldsymbol{L}_{DQ} \cdot \boldsymbol{i}_{DQ} \tag{2.57}$$

where:
$\boldsymbol{v}_{DQ} = (v_{ds}, v_{qs}, v_{dr}, v_{qr})^T$ and $\boldsymbol{i}_{DQ} = (i_{ds}, i_{qs}, i_{dr}, i_{qr})^T$ are voltage and current vectors.
$\boldsymbol{\lambda}_{DQ} = (\lambda_{ds}, \lambda_{qs}, \lambda_{dr}, \lambda_{qr})^T$ is the flux linkage vector.
$\boldsymbol{L}_{DQ} = \begin{pmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{pmatrix}$ is the inductance matrix, where L_s is the stator self-inductance, L_r is the rotor self-inductance, and L_m is the mutual inductance.
$\boldsymbol{R}_{s,DQ} = \begin{pmatrix} R_s & 0 & 0 & 0 \\ 0 & R_s & 0 & 0 \\ 0 & 0 & R_r & 0 \\ 0 & 0 & 0 & R_r \end{pmatrix}$ is the resistance matrix, hwere R_s is the stator resistance, R_r is the rotor resistance.
$\boldsymbol{A}_c = \begin{pmatrix} 0 & -\omega_s & 0 & 0 \\ \omega_s & 0 & 0 & 0 \\ 0 & 0 & 0 & -(\omega_s - \omega_r) \\ 0 & 0 & (\omega_s - \omega_r) & 0 \end{pmatrix}$ is the flux coupling matrix where ω_s is the electrical synchronous speed in rad/s, and ω_r is the rotor electrical speed.

Bearing fault model in induction machine

When bearing fault appears in the machine, some vibration periodic pulses will be generated as a result of the impact among the rolling elements, the bearing raceways, and the cage. The periodic pulses have different characteristic frequency depending on the fault types. Below shows five typical bearing faults and the related characteristic frequencies.

- Cage defect hits outer raceway: $f_{co} = \frac{f_r}{2} \cdot (1 - \frac{d}{D}\cos(\theta))$
- Cage defect hits inner raceway: $f_{ci} = \frac{f_r}{2} \cdot (1 + \frac{d}{D}\cos(\theta))$
- Outer raceway defect hits balls: $f_o = N_b \cdot \frac{f_r}{2} \cdot (1 - \frac{d}{D}\cos(\theta))$
- Inner raceway defect hits balls: $f_i = N_b \cdot \frac{f_r}{2} \cdot (1 + \frac{d}{D}\cos(\theta))$
- Ball defect hits both raceways: $f_b = N_b \cdot \frac{d}{D} \cdot f_r \cdot (1 - \frac{d^2}{D^2}\cos^2(\theta))$

where N_b is the number of balls, f_r is the mechanic rotating frequency of the rotor, and the geometry parameters of the bearing is shown in Fig. 2.17.

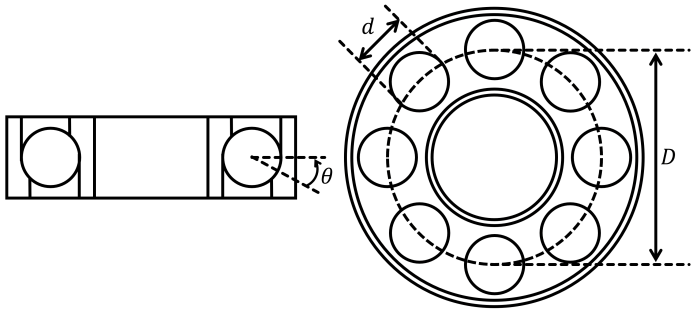


Figure 2.17: Geometry parameters of the bearing.

The periodic pulses caused by vibration will then introduce geometry asymmetry to the machine. Such asymmetry will then change the inductance of the machine. Idealy, the inductance variation should be composed by infinite number of the characteristic frequency harmonics. To simplify the model, we only choose the fundamental frequency component to represent the inductance variation. The impacted inductance is expressed as the follow.

$$L_x^{impacted} = L_x + \Delta L_m \cos(\omega_c t) \tag{2.58}$$

where ΔL_m is the variation magnitude and ω_c is the angular characteristic frequency.

Thus, the new inductance matrix $\boldsymbol{L}_{f,DQ}$ will be updated as followings.

$$\boldsymbol{L}_{f,DQ} = \boldsymbol{L}_{DQ} + \boldsymbol{\Delta L} \tag{2.59}$$

$$\boldsymbol{\Delta L} = \Delta L_m \cos(\omega_c t) \cdot \boldsymbol{M} \tag{2.60}$$

$$\boldsymbol{M} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \tag{2.61}$$

Then, the system model under fault condition could be written as:

$$\boldsymbol{v}_{f,DQ} = \boldsymbol{R}_{DQ} \cdot \boldsymbol{i}_{f,DQ} + \frac{d}{dt}\boldsymbol{\lambda}_{f,DQ} + \boldsymbol{A}_{c1} \cdot \boldsymbol{\lambda}_{f,DQ} \tag{2.62}$$

$$\boldsymbol{\lambda}_{f,DQ} = \boldsymbol{L}_{DQ} \cdot \boldsymbol{i}_{f,DQ} + \boldsymbol{\Delta L} \cdot \boldsymbol{i}_{f,DQ} \tag{2.63}$$

State-Space Model of Bearing Fault in IM

Similar to the ITSC fault in PMSM, bearing fault model in IM could also be re-written to the proposed state-space model, which seperate fault dynamics from the original system model.

$$\text{Basic} \begin{cases} \frac{d}{dt}\boldsymbol{x}_2 = \boldsymbol{A}_{B2} \cdot \boldsymbol{x}_2 + \boldsymbol{B}_{B2} \cdot \boldsymbol{u}_2 + \boldsymbol{E}_{B2} \cdot \boldsymbol{f}_2 \\ \boldsymbol{y}_2 = \boldsymbol{C}_{B2} \cdot \boldsymbol{x}_2 + \boldsymbol{F}_{B2} \cdot \boldsymbol{f}_2 \end{cases} \tag{2.64}$$

$$\text{Fault: } \boldsymbol{f}_1 = \boldsymbol{D}_{F2} \cdot \boldsymbol{y}_2 \tag{2.65}$$

where $\boldsymbol{x}_2 = \boldsymbol{\lambda}_{f,DQ}$, $\boldsymbol{y}_2 = \boldsymbol{i}_{f,DQ}$, $\boldsymbol{u}_2 = \boldsymbol{v}_{f,DQ}$, $\boldsymbol{f}_2 = \boldsymbol{\lambda}_f$. And the parameter matrices are $\boldsymbol{A}_{B2} = -(\boldsymbol{R}_{DQ} \cdot \boldsymbol{L}_{DQ}^{-1} + \boldsymbol{A}_{c1})$, $\boldsymbol{B}_{B2} = \boldsymbol{I}$, $\boldsymbol{E}_{B2} = \boldsymbol{R}_{DQ} \cdot \boldsymbol{L}_{DQ}^{-1}$, $\boldsymbol{C}_{B2} = \boldsymbol{L}_{DQ}^{-1}$, $\boldsymbol{F}_{B2} = -\boldsymbol{L}_{DQ}^{-1}$, and $\boldsymbol{D}_{F2} = \boldsymbol{\Delta L}$.

2.4 Physics-Based Equations for Cyberattacks Targeting Intelligent Electric Drive Systems

With the pervasive utilizations of the digital control units in modern motor drive networks, the potential threats from cyber-domain grow drastically. Here we introduce a general linear model for the cyber attacks in motor drive networks.

2.4.1 General Motor Drive Controller Structure

In order to study both physical faults and cyber-attacks, we proposed a general structure of the motor drive controller as shown in Fig. 2.18. We seperate the physical plant into two subsystems: one is the

original machine dynamics under healthy conditions and the other one is the fault dynamics. Then, the digital controller model is added to the plant model.

Meanwhile, a system monitor is added to the model as well to emulate the traditional system condition monitor and anomaly detector. As shown in Fig. 2.18, y_k is the measurements from the plant and u_k is the control forces from the controller. y_k and u_k are the primary interactions between the physical plants and digital controllers. Such interactions are usually realized by some forms of communication networks. In normal conditions, the communication networks are safe and secure, then $y_k = \hat{y}_k$ and $u_k = \hat{u_k}$. However, when the communication networks is under attacks, $y_k = \hat{y}_k$ and $u_k = \hat{u_k}$ may no longer hold.

Before introduce the attack models, the basic machine model $\boldsymbol{B}$, the fault dynamic model $\boldsymbol{F}$, the digital controller model $\boldsymbol{C}$, and the monitor model $\boldsymbol{D}$ are described in discrete manners. It should be noted the basic machine models and fault dynamic models could be specialized from Eq. (2.46), Eq. (2.47), Eq. (2.64), and Eq. (2.65).

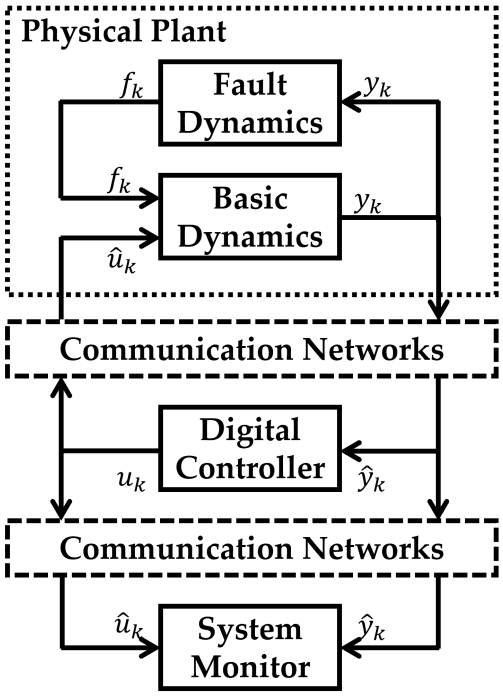


Figure 2.18: General structure of the motor drive controller.

$$\boldsymbol{B}: \begin{cases} \boldsymbol{x}_{k+1} = \boldsymbol{A}_B \boldsymbol{x}_k + \boldsymbol{B}_B \hat{\boldsymbol{u}}_k + \boldsymbol{E}_B \boldsymbol{f}_k + \boldsymbol{G}_B \boldsymbol{w}_k \\ \boldsymbol{y}_k = \boldsymbol{C}_B \boldsymbol{x}_k + \boldsymbol{F}_B \boldsymbol{f}_k + \boldsymbol{H}_B \boldsymbol{w}_k \end{cases} \tag{2.66}$$

$$\boldsymbol{F}: \begin{cases} \boldsymbol{s}_{k+1} = \boldsymbol{A}_F \boldsymbol{s}_k + \boldsymbol{B}_F \boldsymbol{y}_k + \boldsymbol{G}_F \boldsymbol{w}_k \\ \boldsymbol{f}_k = \boldsymbol{C}_F \boldsymbol{s}_k + \boldsymbol{D}_F \boldsymbol{y}_k + \boldsymbol{H}_F \boldsymbol{w}_k \end{cases} \tag{2.67}$$

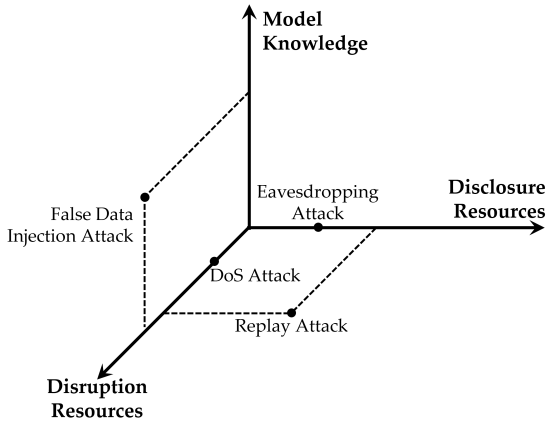
$$\boldsymbol{C}: \begin{cases} \boldsymbol{z}_{k+1} = \boldsymbol{A}_c \boldsymbol{z}_k + \boldsymbol{B}_c \hat{\boldsymbol{y}}_k \\ \boldsymbol{u}_k = \boldsymbol{C}_c \boldsymbol{z}_k + \boldsymbol{D}_c \hat{\boldsymbol{y}}_k \end{cases} \tag{2.68}$$

$$\boldsymbol{D}: \begin{cases} \boldsymbol{d}_k = \boldsymbol{A}_D \boldsymbol{d}_k + \boldsymbol{B}_D \hat{\boldsymbol{u}}_k + \boldsymbol{K}_D \hat{\boldsymbol{y}}_k \\ \boldsymbol{r}_k = \boldsymbol{C}_D \boldsymbol{d}_k + \boldsymbol{D}_D \hat{\boldsymbol{u}}_k + \boldsymbol{E}_D \hat{\boldsymbol{y}}_k \end{cases} \tag{2.69}$$

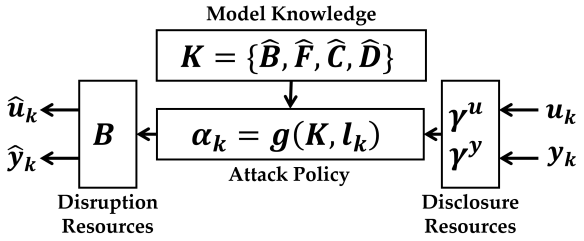
2.4.2 Cyberattack Model Targeting Electric Drive Systems

To model the cyber attacks in motor drive networks, we first classify the attacks into an attack space. Fig. 2.19a shows a conceptual diagram to visualize the attack space. Such attack space has three dimensions:

the adversary's a priori system model knowledge, disclosure, and disruption resources. The a priori model knowledge can be used by the adversary to construct more complex attacks, possibly harder to detect and with more severe consequencies. The disclosure resources enable the adversary to obtain sensitive information about the system during the attack by violating data confidentiality. Note that disclosure resources alone cannot disrupt the system operation. On the other hand, the disruption resources can be used to affect the system operation, which happens for instance when data integrity or availability properties are violated.(Teixeira et al., 2015)



(a) Diagram of the attack space.



(b) Diagram of the adversary model.

Figure 2.19: Diagram of the cyberattack model targeting intelligent electric drive systems.

Based on the above attack space, every cyber attacks could be developed through the adversary model shown in Fig. 2.19b. In this model, the attacks is composed of an attack policy and teh adversary resources i.e., the system model knowledge, the disclosure resources, and the disruption resources. Each of the adversary resources could be mapped to a specific axis of the attack space. $\boldsymbol{K} = \{\hat{\boldsymbol{B}}, \hat{\boldsymbol{F}}, \hat{\boldsymbol{C}}, \hat{\boldsymbol{D}}\}$ is the a priori model knowledge possessed by the adversary; $\boldsymbol{l}_k$ corresponds to the set of sensor and actuator data available to the adversary at time k; $\boldsymbol{\alpha}_k$ is the attack vector at time k that may affect the system behavior using the disruption resources captured by $\boldsymbol{B}$.

2.5 Hybrid Model for Cyberattacks Targeting Electric Drive Systems: Control Information Flow (CIF) Model

Previous sections established the physics-based equation models for intelligent electric drive systems regarding common physical faults and cyberattacks. Such equation-based models are widely used in existing security literature on power and industrial systems including supervisory control and data acquisition (SCADA) systems (Yulia et al., 2016; Zhu & Sastry, 2010), micro-grids (Beheshtaein et al., 2019; Canaan et al., 2020), and industrial control systems (Bhamare et al., 2020). However, while power/control engineers tend to adopt these equations-based system dynamics (Giraldo et al., 2018; Guo et al., 2022; Teixeira

et al., 2015; Yang, Guo, Li, et al., 2020a; Yang, Ye, & Guo, 2022; J. Ye et al., 2021), computer scientists have focused their cyber security research on vulnerabilities of industrial controllers and their communication protocols, such as SRAMs of microcontrollers (MCU) / digital signal processors (DSP) (Skorobogatov & Anderson, 2002), controller area networks (CAN) (Hounsinou et al., 2021; Jafarnejad et al., 2015), and Modbus protocols (Drias et al., 2015).

Therefore, a gap is formed between studies conducted by computer scientists and power/control engineers in this interdisciplinary area. Such a gap will be summarized as follows:

1. Computer scientists have explored the vulnerabilities of specific devices and protocols, such as MCU, DSP, CAN, and Modbus.(Drias et al., 2015; Hounsinou et al., 2021; Jafarnejad et al., 2015; Skorobogatov & Anderson, 2002) However, they do not connect those vulnerabilities to corresponding physical plants, such as inverters for electric machine drives. Therefore, potential impacts on physical plants due to existing cyber vulnerabilities are unclear.

2. Power/control engineers (Giraldo et al., 2018; Teixeira et al., 2015; Yang, Guo, Li, et al., 2020a) usually adopt state-space models to model the plant dynamics and control laws. They assume that attack policies are directly applied to control commands $\boldsymbol{u}$ and system feedbacks $\boldsymbol{y}$, such as the one shown in Fig. 2.20. However, $\boldsymbol{u}$ and $\boldsymbol{y}$ cannot trace back to real-world onboard vulnerable resources that computer scientists explored. Therefore, it is hard to accurately predict the potential cyber-attack impacts on physical plants from real-world cyber threats.

3. For the above two reasons, computer science and power/control engineering research outcomes cannot link to each other, and real-world cyber-threats cannot connect to specific physical systems. This gap becomes a critical obstacle to security and safety with pervasive penetrations of digital control units in modern electric machine drives.

Therefore, this section proposes a new security framework for digitally-controlled electric machine drives to study the system impacts due to security vulnerabilities. The proposed security framework is a hybrid model combining physics-based equations with agent-based models, which promises to connect real-world onboard vulnerable resources with physical systems using a four-layer framework: cyber layer, control layer, physical layer, and impact layer. Fig. 2.20 shows a conceptual diagram for the proposed model and compares with the traditional model. The core of the proposed model is an innovative control information flow (CIF) model in the control layer, which bridges the gap between vulnerable resources in the cyber-layer and state-space models in the physical layer. The CIF model is inspired by the classic information flow model (Austin & Flanagan, 2009; Sabelfeld & Myers, 2003) and taint graph analysis (Ming et al., 2015; Tripp et al., 2009) from informatics researches. It could locate the vulnerable resources in the control variable space and trace the propagation paths from attacked control variables to tainted control laws. Then, accurate predictions of the attack impacts could be achieved by solving the system state-space model associated with the tainted control laws.

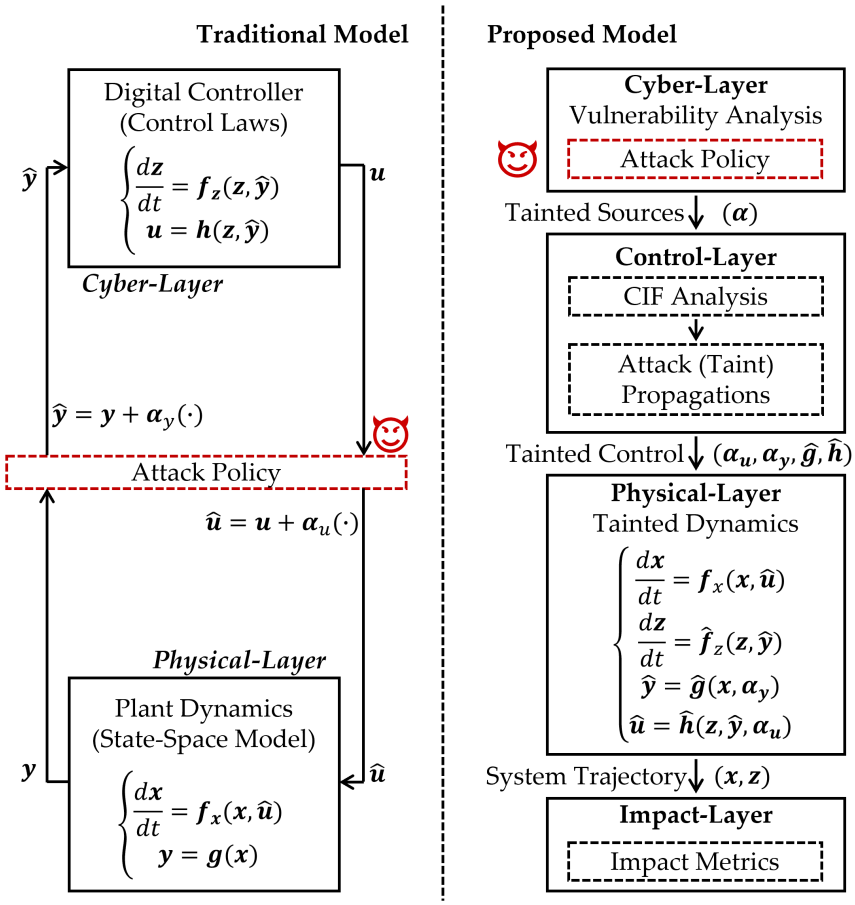


Figure 2.20: Comparison between the traditional impact analysis framework (left) and the proposed security framework (right). The traditional model assumes attacks are directly added to control outputs ($\boldsymbol{u}$) and feedbacks ($\boldsymbol{y}$). Therefore, it is hard for the traditional model to reveal attack propagation and predict attack impacts. The proposed model solves this problem using the CIF model and attack propagation analysis, which could generate detailed tainted control laws and accurate predictions on system behaviors under attack.

2.5.1 Hybrid Security Framework for Digitally-Controlled Electric Machine Drives

The proposed security framework for electric machine drives aims to fill the gap among existing literature on cyber-physical security research from different communities and solve the practical issue of connecting real-world cyber-attacks to power electronics system analysis. The proposed model consists of four layers: cyber layer, control layer, physical layer, and impact layer. Fig. 2.20 shows a diagram of the proposed four-layer security framework for electric machine drives. The cyber layer formulates the vulnerable resources

and potential cyber-attacks into a list of tainted sources ($\boldsymbol{\alpha}$). The control layer tracks the propagation of these tainted sources and generates the tainted control laws ($\boldsymbol{\alpha}_u$, $\boldsymbol{\alpha}_y$, $\hat{\boldsymbol{g}}$, $\hat{\boldsymbol{h}}$). The physical layer maps the tainted control laws to the original system dynamics and calculates the state trajectories under the attack. Finally, the impact layer defines two metrics evaluating attack impacts based on the predicted state trajectories.

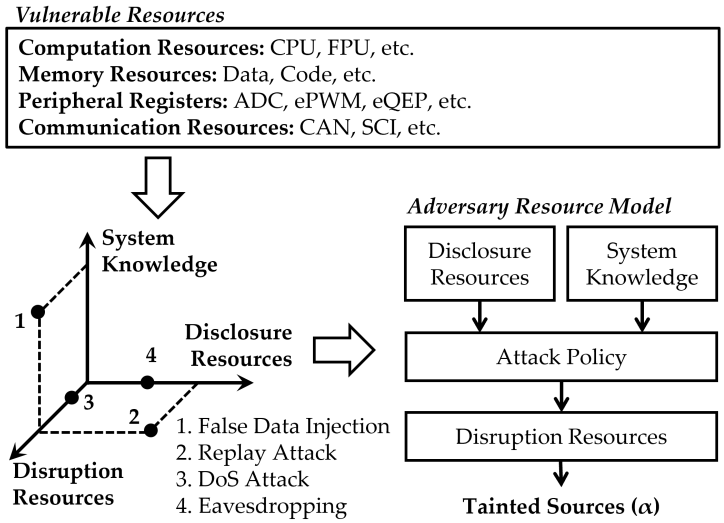


Figure 2.21: Diagram of the cyber layer based on the adversary resource model.

Cyber Layer

Existing literature have explored vulnerable resources in MCUs and DSPs, such as SRAMs (Skorobogatov & Anderson, 2002), CAN (Hounsinou et al., 2021; Jafarnejad et al., 2015), and Modbus protocols (Drias et al., 2015). These vulnerable resources suffer from various cyber-attacks, such as buffer overflows, man-in-the-middle attacks, and false data injections (FDI). The cyber layer adopts the adversary resource model to formulate these attacks to a list of tainted sources in the controller ($\boldsymbol{\alpha}$). Fig. 2.21 shows a diagram of the cyber layer, which first categorizes vulnerable resources into system knowledge, disclosure resources, and disruption resources. The system knowledge includes critical information about the controllers, such as variable locations, control-law-related function instances, and communication protocols. The disclosure and disruption resources are defined by the attacker's access permissions of specific onboard resources. Then, the adversary resource model connects the system knowledge and disclosure resources to the disruption resources using a formulated attack policy and generates a list of tainted sources. Such tainted sources will then be mapped to the CIF model in the control layer as the starting point of the attack propagation analysis. For example, suppose the address for the inverter phase A current ADC offset variable (x_{A0}) is 0x00C000 - 0x00C001 (the data type for the offset variable is int32), and there is a vulnerability report stating that there is a potential threat of buffer-overflows at 0x00C000 - 0x00CFFF. The cyber layer will first determine that x_{A0} is the only variable stored in oxooCooo - oxooCFFF and map

x_{A0} to the CIF model as a tainted source. Meanwhile, the tainted x_{A0} will be denoted as $\hat{x}_{A0} = x_{A0} + \alpha$, since x_{A0} could be falsely modified by buffer-overflows. Then, the propagation and impacts of this threat will be analyzed in the control layer.

Control Layer

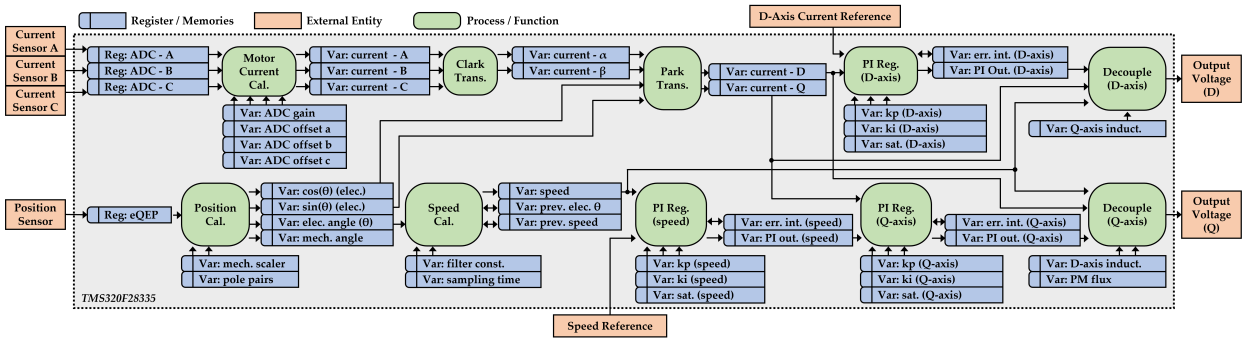


Figure 2.22: Diagram of the control information flow (CIF) model for a PMSM drive with field-oriented-control (FOC). Processes and functions (green blocks) are the backbones of the CIF model, which consist of the calculations and operations. The memory and register data (blue blocks) are categorized into process/function inputs, outputs, and parameters. For each process/function, inputs are on the left, outputs are on the right, and parameters are on the bottom. External entities (orange blocks) represent the peripheral devices and information from outside the controller. The arrows denote the information flow directions. (The PWM modulation processes are neglected because this example does not include the power electronics switching model.)

The control layer is the core of the proposed security framework. It is based on an innovative CIF model, which could trace the tainted sources from the cyber-layer and develop tainted control laws. Such tainted control laws will be the critical bridge between real-world cyber threats in the cyber layer and system dynamics in the physical layer.

Existing literature on cyber-physical security of control systems commonly adopted the framework proposed in (Teixeira et al., 2015). Fig. 2.20 shows a comparison diagram between the traditional analysis framework and the proposed security framework in analyzing attack impacts on electric machine drives. The traditional control model assumes that attack policies are directly added to the feedback signals $\boldsymbol{y}$ and control outputs $\boldsymbol{u}$. Eq. (2.70) and Eq. (2.71) are common models for these attacks, A common attack model in shown in Fig. 2.20, where $\boldsymbol{\alpha}_y$ and $\boldsymbol{\alpha}_u$ are pre-defined functions based on the attack policies. Nevertheless, real-world cyber-attacks include complex interactions among different resources and procedures, and this traditional model is insufficient to capture these interactions and analyze attack impacts and propagation.

$$\hat{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{\alpha}_y(\cdot) \tag{2.70}$$

$$\hat{\boldsymbol{u}} = \boldsymbol{u} + \boldsymbol{\alpha}_u(\cdot) \tag{2.71}$$

As shown in Fig. 2.20, the control layer adopts a proposed control information flow (CIF) model, to fill the gap between real-world attacks and control system analysis. The CIF model categorizes different onboard resources into processes/functions, registers/memories, and external entities.

- **Processes/Functions:** Recent developments in control firmware tend to be modularized, which means control functions are packed into flexible modules. These modules range from ADC data conversions to proportional-integral(PI) controllers. As these modules determine the overall structures of the digital controller, they will be the backbone of the CIF model and be constructed first. In DSPs and MCUs, these processes and functions are located in memory program sectors. When the attackers explore vulnerabilities in these resources, such as back-door instructions, they can maliciously modify control logic.

- **Registers/Memories:** There are three associated data types for each process/function: inputs, outputs, and parameters. These data represent the information flowing in and out of each process/function and are stored in registers and memories (data sector). Due to a lack of encryption and authentication, these data are vulnerable to various cyber-attacks, such as buffer overflows and false data injections. In the CIF model, inputs, outputs, and parameters are defined as registers/memories block with different locations with respect to their processes/functions. For each process/function, its inputs are on the left, outputs are on the right, and parameters are on the bottom.

- **External Entities:** Besides processes/functions and registers/memories, various peripheral units are associated with digital controllers' interactions with external information, such as sensor signals. The CIF model defines these external information sources as external entities. In addition, the output control commands, such as pulse-width modulation(PWM) voltages, are also defined as external entities.

Then, the CIF model maps these resources to different control elements (function blocks and variables), such as motor current calculations and Park transformations. In the end, the information flow direction among these resources is determined and denoted by arrows.

For example, Fig. 2.22 shows the CIF model based on the field-oriented-control (FOC) for a permanent magnet synchronous machine (PMSM). In DSP/MCU-controlled electric machine drive systems, the interrupt service routine (ISR) is the primary manner to implement the control algorithms. The ISR consists of processes and functions which interact with peripheral registers and memories. The three types of resources are denoted as processes/functions (green blocks), registers/memories (blue blocks), and external entities (orange blocks). In DSP/MCU-controlled electric machine drive systems, the interrupt service routine (ISR) is the primary manner to implement the control algorithms. Therefore, their execution sequences determine the arrangement of processes/functions. Then, inputs, outputs, and predefined parameters are aligned with each process/function. If a process/function reads and writes to the same variable in one control cycle, this variable is set on the right of this process/function. Additionally, the CIF model only includes control-related variables because this model primarily focuses on the attack

impacts on control performances. In the end, the peripheral units are located outside processes/functions and registers/memories.

One of the key benefits of the CIF model is that the model complexity is adaptive to different types of cyber threats. For example, if the explored vulnerability focuses on control logic, the PWM modulations could be neglected and approximated by simple voltage outputs, such as the one shown in Fig. 2.22. Meanwhile, if the targeted attack focuses on the speed calculation procedures in Fig. 2.22, extra details like the digital low-pass filter in speed calculations could be added to the CIF model by adding extra registers/memories blocks. With this benefit, the CIF model could be flexible and adaptive to different cyber vulnerabilities. In Fig. 2.22, the CIF model neglects the PWM-related processes and defines the D- and Q-Axis voltages as the system outputs.

After establishing the CIF model, the control layer locates the tainted sources ($\boldsymbol{\alpha}$) from the cyber layer in the CIF model. Then, the propagation of tainted sources could be resolved by tracking the flowing paths of each source. (Fig. 2.23 and Fig. 2.24) Once determining the propagation paths, tainted control laws ($\boldsymbol{\alpha}_u$, $\boldsymbol{\alpha}_y$, $\hat{\boldsymbol{g}}$, $\hat{\boldsymbol{h}}$) could be extracted and fed to the physical layer.

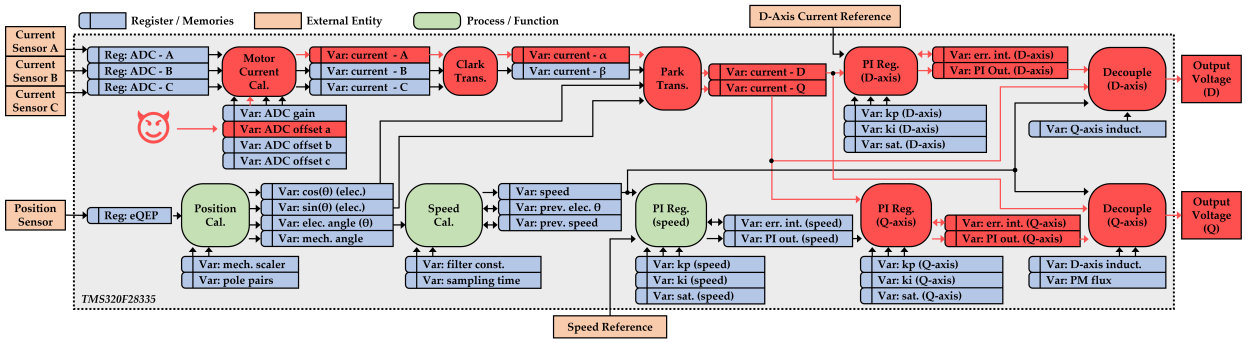


Figure 2.23: Diagram of the attack propagation tracing for case 1. The tainted source (phase-A ADC offset variable) marks the starting point of the propagation paths. The tainted processes, functions, and variables are colored red.

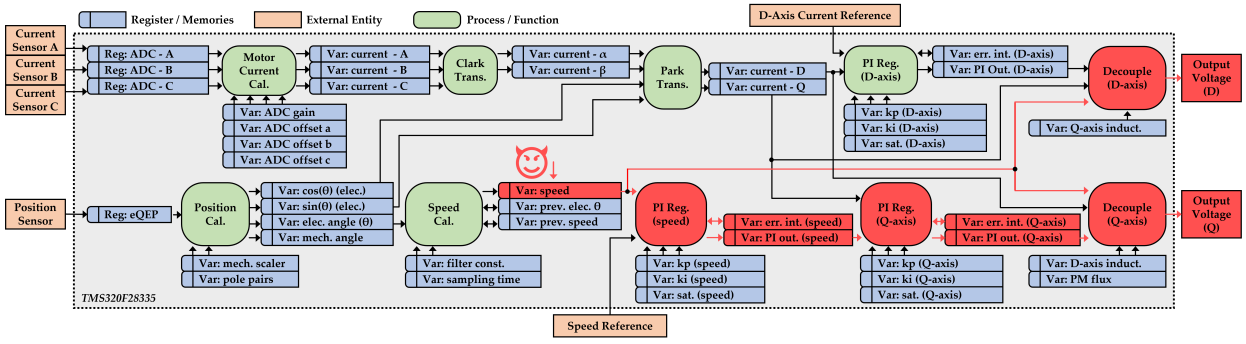


Figure 2.24: Diagram of the attack propagation tracing for case 2. The tainted source (calculated speed variable) marks the starting point of the propagation paths. The tainted processes, functions, and variables are colored red.

Physical Layer

The physical layer consists of the state-space model of the electric machine drive. For example, the dynamics of a PMSM drive in DQZ reference frame could be modeled as Eq. (2.72)-Eq. (2.74), where ω_m, i_d, i_q are the speed, d-axis current, and q-axis current; J, R_m, T_L are the inertia, damping coefficient, and load torque; R_s, L_s, λ_{PM} are the stator resistance, inductance, and magnet flux linkage.

$$\frac{d\omega_m}{dt} = -\frac{R_m}{J}\omega_m + \frac{\lambda_{PM}}{J}i_q - \frac{T_L}{J} \tag{2.72}$$

$$\frac{di_d}{dt} = (-\frac{R_s}{L_s}i_d + \omega_m i_q + \frac{1}{L_s}v_d) \cdot \omega_b \tag{2.73}$$

$$\frac{di_q}{dt} = (-\frac{R_s}{L_s}i_q - \omega_m i_d - \omega_m\frac{\lambda_{PM}}{L_s} + \frac{1}{L_s}v_q) \cdot \omega_b \tag{2.74}$$

v_d and v_q are the d- and q-axis voltage inputs, which are derived from the FOC laws shown in Eq. (2.75)-Eq. (2.82), where I_d, I_q, Ω_m are the references of d-axis current, q-axis current and motor speed; z_d, z_q, z_m are the error integration from PI regulators; K_{mp}, K_{mi}, K_{dp}, K_{di}, K_{qp}, K_{qi} are the PI control coefficients. All variables in Eq. (2.72)-Eq. (2.82) are in per-unit manner where ω_b is the base value for synchronous speed.

$$\frac{dz_m}{dt} = \Omega_m - \omega_m \tag{2.75}$$

$$\frac{dz_d}{dt} = I_d - i_d \tag{2.76}$$

$$\frac{dz_q}{dt} = I_q - i_q \tag{2.77}$$

$$I_q = K_{mp}(\Omega_m - \omega_m) + K_{mi} \cdot z_m \tag{2.78}$$

$$u_d = K_{dp}(I_d - i_d) - K_{di} \cdot z_d \tag{2.79}$$

$$u_q = K_{qp}(I_q - i_q) + K_{qi} \cdot z_q \tag{2.80}$$

$$v_d = u_d - \omega_m L_s i_q \tag{2.81}$$

$$v_q = u_q + \omega_m L_s i_d + \omega_m \lambda_{PM} \tag{2.82}$$

Then, the physical layer will replace the original control laws, i.e., Eq. (2.75)-Eq. (2.82), with the tainted control laws from the control layer and solve the close-loop system to get the system state trajectories under specific attacks.

Impact Layer

After acquiring the system state trajectories under specific attacks, the attack impacts could be evaluated by the following definitions of attack properties and system metrics.

Definition 1. *Given a continuous state space $\boldsymbol{X} \subseteq \mathbb{R}^n$, suppose the original closed-loop attack-free system $\dot{\boldsymbol{x}} = f(\boldsymbol{x})$ has an equilibrium point $\boldsymbol{X_e}$, and there exists a set $\mathbb{B}^n$, which satisfies: (1) $\mathbb{B}^n \subseteq \mathbb{R}^n$; (2) $\boldsymbol{X_e} \subseteq$*

$\mathbb{B}^n$; (3) for any initial point $\boldsymbol{X_0} \subseteq \mathbb{B}^n$, the state space $\boldsymbol{X}$ will eventually converge to the equilibrium point $\boldsymbol{X_e}$. Meanwhile, suppose the tainted closed-loop system $\dot{\boldsymbol{x}} = \hat{f}(\boldsymbol{x})$ has a stable equilibrium point $\boldsymbol{X_a}$ and let T_0 be the attack-initiating time and T be the time when the attack could be detected and removed. Then, the attack is (1) "minor" if $||\boldsymbol{X_e} - \boldsymbol{X_a}|| \leq \delta$, δ is a small positive number; (2) "stable" if $||\boldsymbol{X_e} - \boldsymbol{X_a}|| > \delta$ and $\boldsymbol{X_a} \subseteq \mathbb{B}^n$; (3) "drastic" if $\boldsymbol{X_a} \nsubseteq \mathbb{B}^n$ and $\boldsymbol{X}(T) \subseteq \mathbb{B}^n$; (4) "unstable" if $\boldsymbol{X_a} \nsubseteq \mathbb{B}^n$ and $\boldsymbol{X}(T) \nsubseteq \mathbb{B}^n$.

Remark 1. *If $\boldsymbol{X_e}$ is a globally stable equilibrium point, i.e., $\mathbb{B}^n = \mathbb{R}^n$ and $\boldsymbol{X_a}$ is a stable equilibrium point, all attacks will be "minor" or "stable".*

Remark 2. *If the tainted closed-loop system $\dot{\boldsymbol{x}} = \hat{f}(\boldsymbol{x})$ has an unstable equilibrium point $\boldsymbol{X_a}$, the attack is "unstable".*

Definition 2. *Define the first impact index $\mathcal{I}_1$ as the cost required to steer the system states back to origin, i.e.,*

$$\mathcal{I}_1 = \sqrt{\int_T^\infty ||\boldsymbol{x}(t) - \boldsymbol{X_e}||^2 dt} \tag{2.83}$$

Definition 3. *Define the second impact index $\mathcal{I}_2$ as the maximum system deviation from the original equilibrium point, i.e.,*

$$\mathcal{I}_2 = \max\{||\boldsymbol{x}(t) - \boldsymbol{X_e}|| : t \in [T_0, T]\} \tag{2.84}$$

Remark 3. *If the attack is "unstable", $\mathcal{I}_1$ will not be convergent.*

According to the **Definition 2** and **Definition 3**, the first impact index reflects how hard the system could recover from the attack, and the second impact index reflects how much disturbance the attack could cause to the system.

Based on the proposed security framework, a systematic attack impact analysis procedure is summarized as follows.

- Step 1: Cyber Layer - Locate the system's vulnerable resources, formulate the attack policy, and generate the tainted sources.
- Step 2: Control Layer - Map the taint sources to the CIF model, track the attack propagation paths, and establish the tainted control laws.
- Step 3: Physical Layer - Replace the original control laws with tainted control laws, solve the tainted closed-loop state-space model, and calculate the state trajectories under attack.
- Step 4: Impact Layer - Evaluate attack impacts and calculate impact metrics.

2.5.2 Impact Analysis of Electric Machine Drives Due to Cyber-Attacks: Case Studies

This section forms two case studies to demonstrate the impact analysis for cyber-attacks on electric machine drives using the proposed security framework. These case studies include an FDI attack on a motor current sensor offset variable and another FDI attack on the calculated motor speed feedback variable. The reason for selecting these two scenarios to form these case studies could be summarized as follows.

1. The FDI attack is one of the most common cyber-attacks, which could compromise and gain control of electric machine drives.

2. Attacks on the motor current sensor offset variable and calculated speed feedback represent two primary attack targets in the close-loop controller for electric machines: the inner current loop with fast responses and the outer speed loop with slow responses.

This section will demonstrate the analysis procedure from vulnerable resources to tainted control laws. The following section will show the results of the predicted state trajectories under attacks and calculated impact metrics. Each case study includes details from tainted variables to tainted control laws.

Case 1: FDI attack on the motor current offset variable

In practical motor drive controllers, current offset variables are critical to compensate for the current sensors' zero drift issues. Most controllers calculate such offset variables during the initialization process. After initializations, these offsets will maintain constant. However, as these offsets are stored in the memory data sections, multiple attacks could access these variables and maliciously modify them, such as buffer-overflow attacks and FDI attacks. This case study considers a scenario where the motor phase A current sensor offset variable is under an FDI attack. According to the proposed security framework, the tainted variable and attack policy is shown in Eq. (2.85), where $x_{offsetA}$ and $\hat{x}_{offsetA}$ is the original and attacked motor phase A current offset variables; α is the attack coefficient.

$$\hat{x}_{offsetA} = x_{offsetA} + \alpha \tag{2.85}$$

After mapping Eq. (2.85) to the CIF model, Fig. 2.23 shows the attack propagation path. Then tainted control laws could be extracted by substuting Eq. (2.85) to each process/function along the propagation path.

1. Motor Current Calculation Process:

$$\hat{i_a} = i_a - k_{adc} \cdot \alpha \tag{2.86}$$

   where k_{adc} is the ADC convertion coefficient.

2. Clark Transformation Process:

$$\hat{i}_\alpha = i_\alpha - \frac{2}{3} k_{adc} \cdot \alpha \tag{2.87}$$

3. Park Transformation Process:

$$\hat{i}_d = i_d - \frac{2}{3} k_{adc} \cdot \cos\theta \cdot \alpha \tag{2.88}$$

$$\hat{i}_q = i_q + \frac{2}{3} k_{adc} \cdot \sin\theta \cdot \alpha \tag{2.89}$$

where θ is the electric angle per unit.

4. D-Axis Current PI Regulator Process: As the PI regulator includes the error integral term, the D-axis control state equation is changed as the follow.

$$\frac{d\hat{z}_d}{dt} = I_d - \hat{i}_d = I_d - (i_d - \frac{2}{3} k_{adc} \cos\theta \cdot \alpha) \tag{2.90}$$

Meanwhile, the PI regulator output will change accordingly.

$$\hat{u}_d = K_{dp}(I_d - (i_d - \frac{2}{3} k_{adc} \cos\theta \cdot \alpha)) - K_{di} \cdot \hat{z}_d \tag{2.91}$$

5. Q-Axis Current PI Regulator Process:

$$\frac{d\hat{z}_q}{dt} = I_q - \hat{i}_q = I_q - (i_q + \frac{2}{3} k_{adc} \sin\theta \cdot \alpha) \tag{2.92}$$

$$\hat{u}_q = K_{qp}(I_q - (i_q + \frac{2}{3} k_{adc} \sin\theta \cdot \alpha)) - K_{qi} \cdot \hat{z}_q \tag{2.93}$$

6. D-Axis and Q-Axis Feedforward Decoupling Process: this process is the final stage of the FOC control and generates the d- and q-axis voltage commands.

$$\hat{v}_d = \hat{u}_d - \omega_m L_s (i_q + \frac{2}{3} k_{adc} \sin\theta \cdot \alpha) \tag{2.94}$$

$$\hat{v}_q = \hat{u}_q + \omega_m (L_s (i_d - \frac{2}{3} k_{adc} \cos\theta \cdot \alpha) + \lambda_{PM}) \tag{2.95}$$

Stacking the above results will generate tainted control laws, which are shown in Eq. (2.96)-Eq. (2.103).

Then, the physical layer will replace Eq. (2.75)-Eq. (2.82) with Eq. (2.96)-Eq. (2.103) and solves the state trajectories under attacks.

$$\frac{dz_m}{dt} = \Omega_m - \omega_m \tag{2.96}$$
$$\frac{d\hat{z}_d}{dt} = I_d - (i_d - \frac{2}{3}k_{adc} \cdot \cos\theta \cdot \alpha) \tag{2.97}$$
$$\frac{d\hat{z}_q}{dt} = I_q - (i_q + \frac{2}{3}k_{adc} \cdot \sin\theta \cdot \alpha) \tag{2.98}$$
$$I_q = K_{mp}(\Omega_m - \omega_m) + K_{mi} \cdot z_m \tag{2.99}$$
$$\hat{u}_d = K_{dp}(I_d - (i_d - \frac{2}{3}k_{adc} \cdot \cos\theta \cdot \alpha)) - K_{di} \cdot \hat{z}_d \tag{2.100}$$
$$\hat{u}_q = K_{qp}(I_q - (i_q + \frac{2}{3}k_{adc} \cdot \sin\theta \cdot \alpha)) + K_{qi} \cdot \hat{z}_q \tag{2.101}$$
$$\hat{v}_d = \hat{u}_d - \omega_m L_s(i_q + \frac{2}{3}k_{adc} \cdot \sin\theta \cdot \alpha) \tag{2.102}$$
$$\hat{v}_q = \hat{u}_q + \omega_m L_s(i_d - \frac{2}{3}k_{adc} \cdot \cos\theta \cdot \alpha) + \omega_m \lambda_{PM} \tag{2.103}$$

Case 2: FDI attack on the motor speed feedback variable

Besides current offset variables, the calculated speed feedback is also a vulnerable target of malicious attacks. For example, the Stuxnet worm compromised the industrial control system by manipulating the rotating speeds of industrial motor drives (Karnouskos, 2011). Suppose the attack policy is the same as Eq. (2.85), which is shown in Eq. (2.104), where ω_m and $\hat{\omega}_m$ is the original and attacked motor speed feedback variables.

$$\hat{\omega}_m = \omega_m + \alpha \tag{2.104}$$

After mapping Eq. (2.104) to the CIF model, Fig. 2.24 shows the attack propagation path. Then tainted control laws are derived as follows.

1. Speed PI Regulator Process:

$$\frac{d\hat{z}_m}{dt} = \Omega_m - \hat{\omega}_m = \Omega_m - (\omega_m + \alpha) \tag{2.105}$$
$$\hat{I}_q = K_{mp}(\Omega_m - (\omega_m + \alpha)) + K_{mi} \cdot \hat{z}_m \tag{2.106}$$

2. Q-Axis Current PI Regulator Process:

$$\frac{d\hat{z}_q}{dt} = \hat{I}_q - i_q \tag{2.107}$$
$$\hat{u}_q = K_{qp}(\hat{I}_q - i_q) + K_{qi} \cdot \hat{z}_q \tag{2.108}$$

3. D-Axis and Q-Axis Feedforward Decoupling Process:

$$\hat{v}_d = u_d - (\omega_m + \alpha)L_s i_q \tag{2.109}$$
$$\hat{v}_q = \hat{u}_q + (\omega_m + \alpha)L_s i_d + (\omega_m + \alpha)\lambda_{PM} \tag{2.110}$$

Stacking above results will generate tainted control laws, which are shown in Eq. (2.111)-Eq. (2.118).

$$\frac{d\hat{z}_m}{dt} = \Omega_m - \hat{\omega}_m = \Omega_m - (\omega_m + \alpha) \tag{2.111}$$
$$\frac{dz_d}{dt} = I_d - i_d \tag{2.112}$$
$$\frac{d\hat{z}_q}{dt} = \hat{I}_q - i_q \tag{2.113}$$
$$\hat{I}_q = K_{mp}(\Omega_m - (\omega_m + \alpha)) + K_{mi} \cdot \hat{z}_m \tag{2.114}$$
$$u_d = K_{dp}(I_d - i_d) - K_{di} \cdot z_d \tag{2.115}$$
$$\hat{u}_q = K_{qp}(\hat{I}_q - i_q) + K_{qi} \cdot \hat{z}_q \tag{2.116}$$
$$\hat{v}_d = u_d - (\omega_m + \alpha)L_s i_q \tag{2.117}$$
$$\hat{v}_q = \hat{u}_q + (\omega_m + \alpha)L_s i_d + (\omega_m + \alpha)\lambda_{PM} \tag{2.118}$$

Then, the physical layer will replace Eq. (2.75)-Eq. (2.82) with Eq. (2.111)-Eq. (2.118) and solves the state trajectories under attacks.

2.5.3 Experiment Validations

This section will continue with the impact analysis in section III. The physical layer solves the tainted system dynamics and generates system state trajectories under attack. Meanwhile, this section will also provide hardware experiment results to support the analysis results from the proposed framework.

Fig. 2.25 and TABLE 2.5 show a picture and detail specifications of the experiment platform. The platform adopts a 1.5kW PMSM and sets the operation speed at 1000 rpm. The FOC algorithms are implemented in a TMS320F28335 MCU from Texas Instruments. Both case studies assume that the attack lasts 1 second before the system detects and clears it. In addition, impact metrics are automatically calculated once the attack is removed.

Table 2.5: Specifications of the experiment platform.

Rated Power	1.5 kW	Stator Resistance	0.4050 Ω
Rated Current	8.2 A	Stator Inductance	0.0024 H
DC Bus Voltage	200 V	Magnet Flux Linkage	0.0599 Wb
Rated Frequency	250 Hz	Number of Pole Pairs	5
Control Frequency	10 kHz	Motor Inertia	3.10e-4 kgm^2

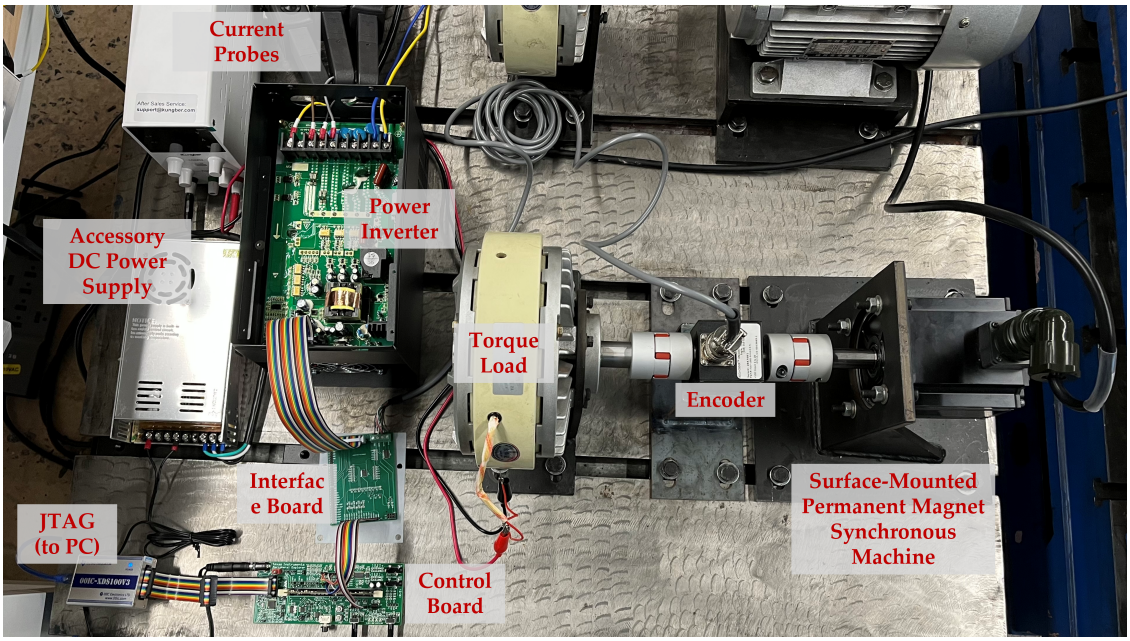


Figure 2.25: Picture of the hardware experiment platform with a PMSM drive.

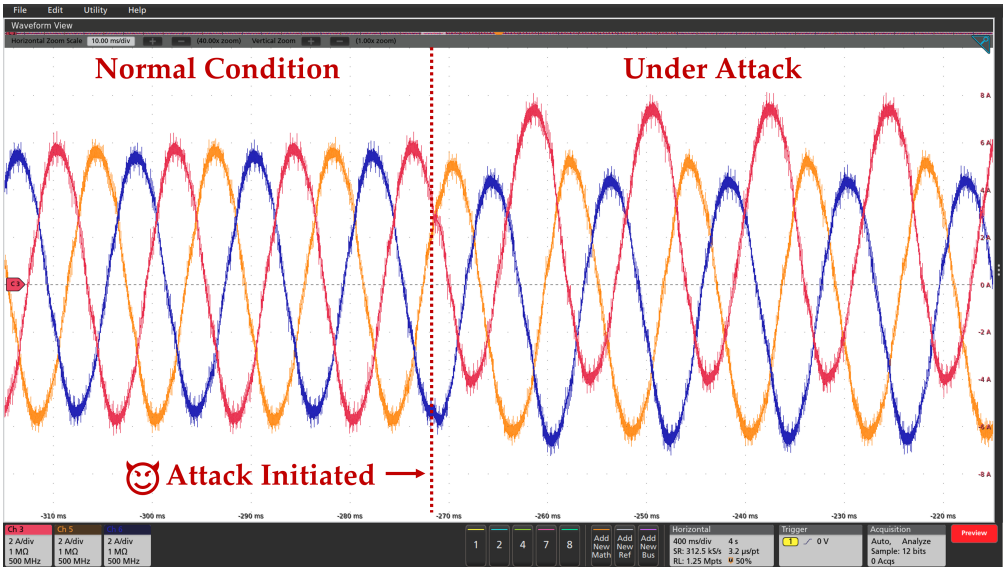


Figure 2.26: Three-phase motor line current waveforms of the PMSM with an FDI attack on phase-A ADC offset variable (case 1). The attack policy follows $\hat{s} = s + \alpha$ with $\alpha = 0.1$. The dashed line marks the attack initiation point. The attacked system shows unbalanced three-phase currents as one of the phase feedback is maliciously biased.

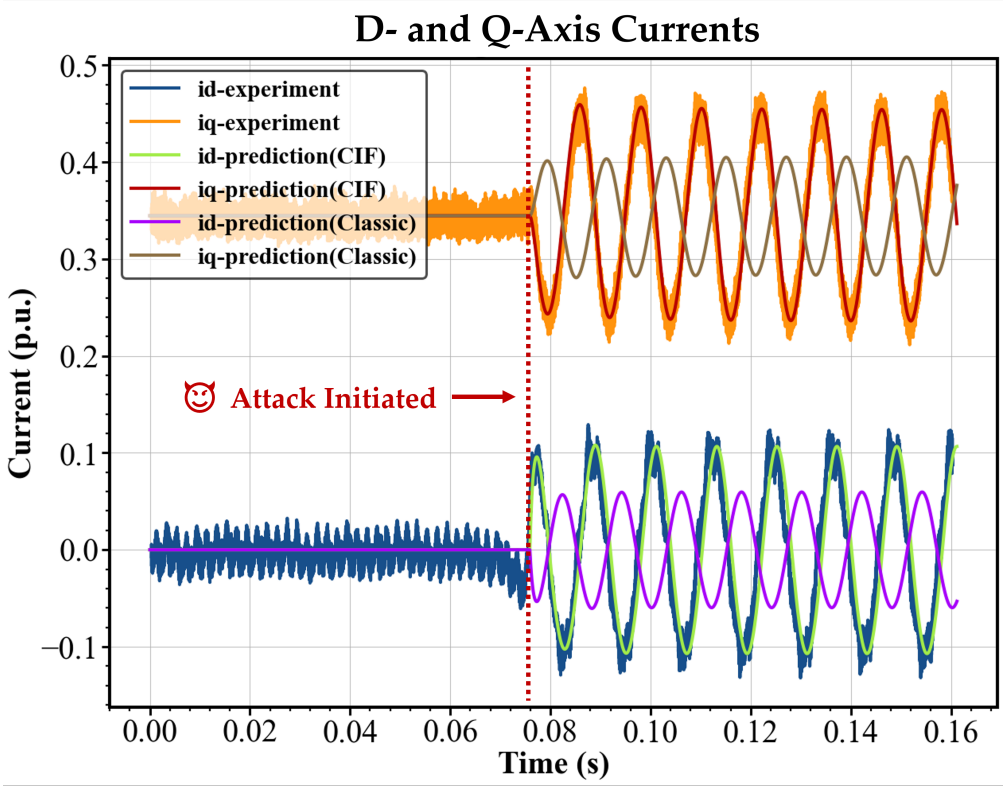


Figure 2.27: D- and Q-Axis current waveforms from the experiment, the CIF model prediction, and the traditional analysis predictions with no extra information of case 1 attack ($\alpha = 0.1$). The experiment results show that the attack causes oscillations in d- and q-axis currents. The CIF modle could predict the system behaviors under attack. However, the classic model shows different results, because if there is no extra information for the classic model, it will directly substitue the attack policy to the current feedback, i.e. $\hat{i}_a = i_a + \alpha$, while the actual tainted current feedback is $\hat{i}_a = k_{adc} * (i_{a,adc} - (i_{a,offset} + \alpha))$.

2.5.4 Case 1: FDI attack on the motor current offset variable

Fig. 2.26 shows the waveforms of the motor line currents when the attack is triggered. The attack policy is shown in Eq. (2.85) where $\alpha = 0.1$. As shown in Fig. 2.26, the attack caused unbalanced currents in the motors, and such unbalance then led to oscillated torque and motor vibrations. After transforming the raw current waveforms to the DQZ reference frame, Fig. 2.27 shows the D- and Q-axis current waveforms. Additionally, Fig. 2.27 also adds the predicted D- and Q- axis currents according to the tainted state trajectories from section III for comparison. As suggested in Fig. 2.27, the predicted results have high accuracy referring to the experiment results.

In addition, Fig. 2.27 also adds the predicted results by the traditional analysis framework, which assumes that the attack policy is directly added to the phase current feedback signal. Therefore, if no extra information is provided to the traditional analysis, there will be no details of attack propagation

from ADC calculations to voltage outputs. It means that the traditional analysis directly treats $x_{offsetA}$ in Eq. (2.85) as the current feedback instead of the offset variable. Therefore, the traditional analysis may generate different results, such as the ones shown in Fig. 2.27.

Then, the attack impact metrics associated with different α are shown in Table. 2.6. As the attack-initiating time is considerably random, the initial phase angles of each experiment are different, which leads to slight variations in the calculated impact metrics. Therefore, the experiment attack impact index in Table. 2.6 is the average value among five independent experiments. The results suggest that the CIF model could accurately predict the attack impact with different attack coefficients.

Case 2: FDI attack on motor speed feedback variable

Fig. 2.28 shows the waveforms of motor line currents when the attack is triggered. Because the attack targets the speed feedback and the mechanical system has slower responses, the current waveforms do not have apparent variations when the attack appears. This characteristic could be one of the reasons the Stuxnet worm could easily compromise the operation speed of industrial motors. Fig. 2.29 shows the transformed current waveforms in the DQZ reference frame associated with the CIF model's predicted trajectories in section III. The results further prove that the proposed CIF model could achieve accurate predictions on the attack impacts.

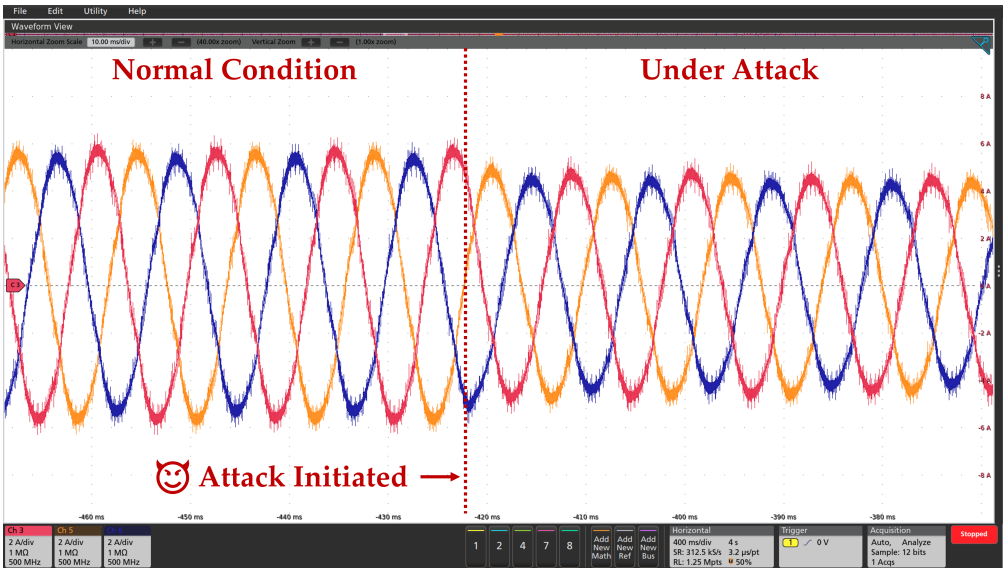


Figure 2.28: Three-phase motor line current waveforms of the PMSM with an FDI attack on speed feedback (case 2). The attack policy follows $\hat{s} = s + \alpha$ with $\alpha = 0.1$. The dashed line marks the attack initiation point. The attacked system shows an obvious change in the current magnitudes.

To further demonstrate the novelty of the proposed CIF model, another similar scenario of the FDI attack on speed feedback (case 2) is discussed as follows. As shown in Fig. 2.24, the attack targets the speed variable instead of the speed calculation process. In the traditional model, these two scenarios should be equivalent since both attacks target the speed feedback. However, the speed calculation process usually

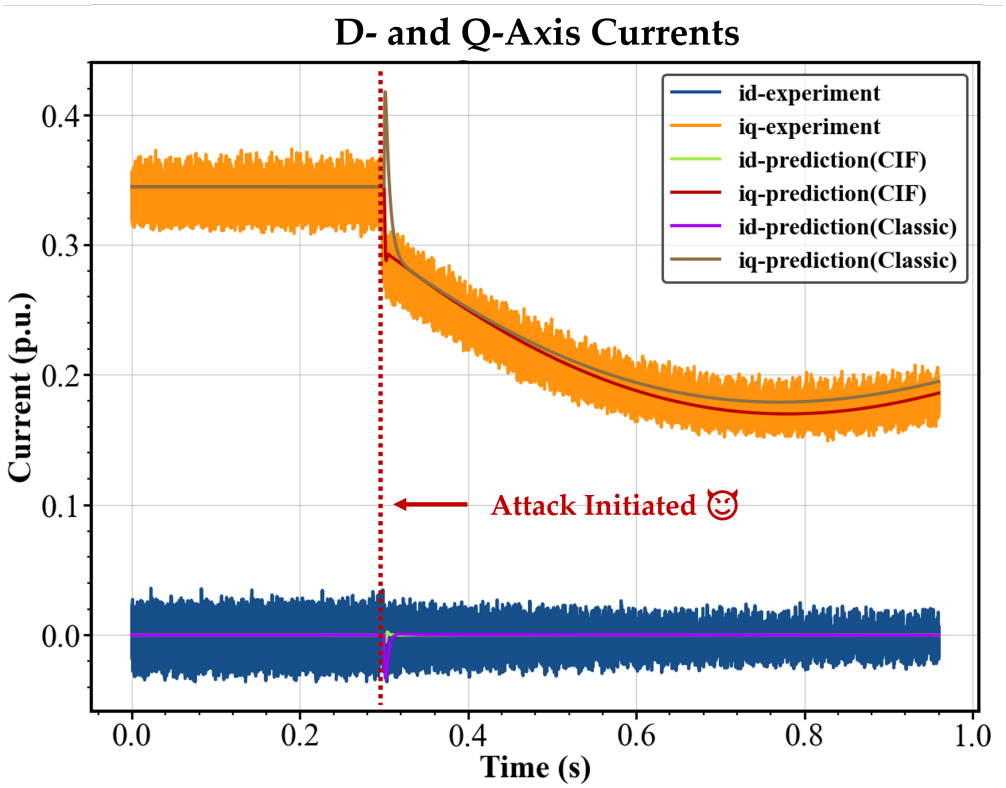


Figure 2.29: D- and Q-Axis current waveforms from the experiment, the CIF model prediction, and the traditional analysis framework predictions of case 2 attack ($\alpha = 0.1$). The experiment results show that the attack causes i_q deviating the original steady-state. The CIF model could accurately predict this impact, while the classic model generates slightly different results with overshoots at the attack-initiating point.

has a small low-pass filter in practical systems. Eq. (2.119) shows the filter calculation in discrete manner, where K_1 and K_2 is the filter coefficients; $\omega_{m(n)}$ and $\omega_{m(n-1)}$ is the current and previous speed feedback; ω_{mc} is the current calculated speed from encoder signals.

$$\omega_{m(n)} = K_1 \cdot \omega_{mc} + K_2 \cdot \omega_{m(n-1)} \tag{2.119}$$

Because of this filter, the speed calculation process needs to update $\omega_{m(n-1)}$ after every control cycle. Therefore, if the attack is targetting $\omega_{m(n)}$ before the update or is targetting ω_{mc}, the injected bias will integrate at a fast pace. Such an integrated bias then leads to the saturation of the speed PI regulator. The tainted voltage output will ultimately cause unstable motor line currents and trigger the pre-defined over-current protection. Fig. 2.30 shows the current waveforms in such a scenario. In practice, the above example is ubiquitous. Therefore, the proposed security framework demonstrates its advancement over the traditional analysis framework, especially in these practical scenarios. Then, the attack impact metrics

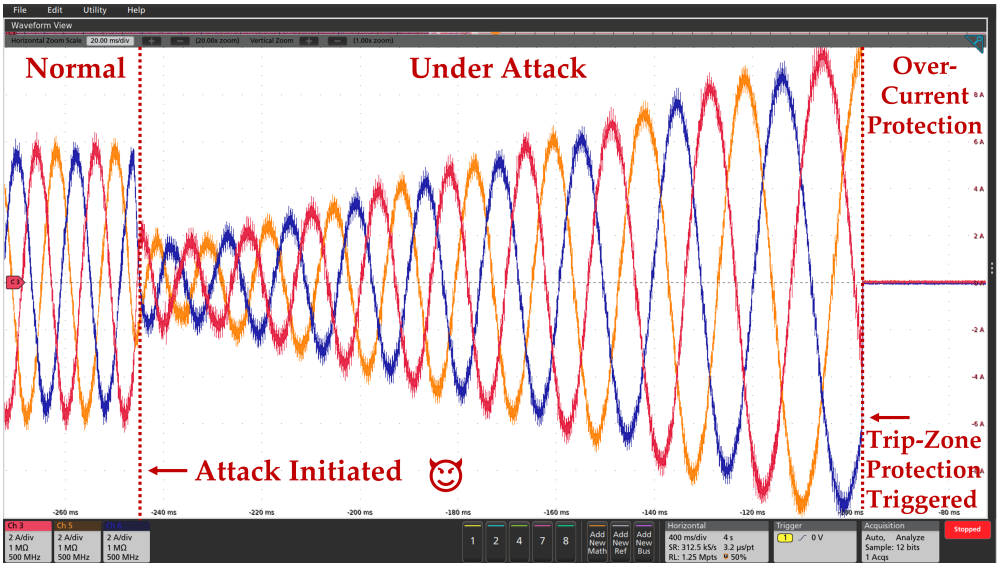


Figure 2.30: Three-phase motor line current waveforms of the PMSM in the extended scenario of case 2, where a FDI attack on ω_{mc} in Eq. (2.119) is implemented. The attack policy follows $\hat{s} = s + \alpha$ with $\alpha = 0.1$. The dashed line marks the attack initiation point and the trip-zone protection point. As the injected bias on ω_{mc} will be integrated every control cycle, the resulting speed bias will quickly diverge, which leads to unstable currents and over-current protection.

Table 2.6: Case 1 Impact Metrics

α	pred. $\mathcal{I}_1$	pred. $\mathcal{I}_2$	expe. $\mathcal{I}_1$	expe. $\mathcal{I}_2$
-0.30	0.0045	0.1924	0.0044	0.1931
-0.25	0.0037	0.1602	0.0036	0.1599
-0.20	0.0028	0.1281	0.0025	0.1273
-0.15	0.0022	0.0960	0.0019	0.0954
-0.10	0.0013	0.0641	0.0011	0.0634
-0.05	0.0008	0.0320	0.0009	0.0329
0.05	0.0007	0.0320	0.0010	0.0314
0.10	0.0017	0.0638	0.0020	0.0628
0.15	0.0019	0.0961	0.0018	0.0968
0.20	0.0029	0.1283	0.0032	0.1281
0.25	0.0033	0.1594	0.0032	0.1605
0.30	0.0051	0.1921	0.0049	0.1912

associated with different α are shown in Table. 2.7. Similar to the previous case study, the experiment attack impact index in Table. 2.7 is the average value among five independent experiments. The results

Table 2.7: Case 2 Impact Metrics

α	pred. $\mathcal{I}_1$	pred. $\mathcal{I}_2$	expe. $\mathcal{I}_1$	expe. $\mathcal{I}_2$
-0.150	0.2955	0.2883	0.2952	0.2891
-0.125	0.2462	0.2402	0.2450	0.2410
-0.100	0.1970	0.1922	0.1977	0.1925
-0.075	0.1477	0.1441	0.1475	0.1426
-0.050	0.0985	0.0961	0.0953	0.9745
-0.025	0.0492	0.0481	0.0491	0.0479
0.025	0.0492	0.0481	0.0487	0.0500
0.050	0.0985	0.0961	0.0989	0.0948
0.075	0.1477	0.1441	0.1490	0.1439
0.100	0.1970	0.1922	0.1951	0.1937
0.125	0.2462	0.2402	0.2471	0.2401
0.150	0.2955	0.2883	0.2941	0.2885

again suggest that the CIF model could accurately predict the impact of this attack with different attack coefficients.

In the end, the overall advancement of the proposed security framework is summarized as follows.

1. The proposed model could establish connections between real-world cyber vulnerabilities and power electronics control frameworks.

2. The proposed analysis framework could predict the attack impacts on electric machine drives due to various vulnerable onboard resources, such as register data and calculation procedures.

3. The proposed CIF model includes detailed information about internal controller structures, which are critical to practical applications.

Chapter 3

Cyber-Physical Testbed with Intelligent Electric Drive Systems

3.1 Hardware-in-the-loop (HIL) Real-Time Simulation Platform

With the increasing complexity of the current intelligent electric drive systems, the traditional single-machine-dominant testing platform can no longer provide all necessary needs, especially regarding cyber-physical system perspectives. Therefore, it is desirable to have a testing platform including exhaustive cyber- and physical domain information. In addition, while a real-world hardware testing platform is an ideal and necessary option, the costs and risks of hardware development should be addressed. Therefore, an alternative approach is the recently arisen hardware-in-the-loop (HIL) real-time (RT) simulation platform.

HIL simulation is a technique that is widely used in the field of engineering for testing and validating control systems. This technique involves connecting a physical system, known as the plant, to a digital simulation environment in real-time. The purpose of this integration is to simulate complex dynamic behaviors of the plant while interacting with a real control system. This allows for the testing of control algorithms and their implementation in hardware without the risk of damaging expensive equipment.

Such approach is an essential tool in the development of control systems for complex, large-scale systems, such as intelligent manufaturing systems, power grids, and automotive systems. These systems are typically characterized by high levels of complexity, with multiple interacting components and dynamic behavior. Developing and analyzing dynamic behaviors for such systems is a challenging task that requires extensive testing and validation. HIL simulation provides a safe and cost-effective means of testing control algorithms before their implementation in the actual system.

The HIL simulation process typically involves three main components: the plant, the control system, and the simulation environment. The plant is the physical system that is being controlled, and it can range from a simple laboratory setup to a complex, real-world system. The control system is the software and hardware that implements the control algorithm, and it is connected to the plant through various sensors

and actuators. The simulation environment is the software that simulates the dynamic behavior of the plant and provides a real-time interface for the control system.

One of the key advantages of HIL simulation is its ability to test control algorithms and system dynamics under realistic conditions. The simulation environment can be programmed to simulate a wide range of operating conditions, including variations in load, environmental conditions, and system faults. This allows for the testing of system behaviors under a range of scenarios, which can help identify potential problems and improve the overall performance of the system.

Another advantage of HIL simulation is its ability to test different algorithms in a safe and controlled environment. In many cases, testing control algorithms in the actual system can be dangerous and expensive. HIL simulation provides a safe and cost-effective means of testing control algorithms, which can ultimately lead to improved system performance and increased efficiency.

In summary, HIL simulation is a powerful technique that is widely used in the field of engineering for testing and validating control systems. It provides a safe and cost-effective means of testing system behaviors before their implementation in the actual system. HIL simulation is particularly useful for testing system behaviors under various scenarios for complex, large-scale electric drive systems, such as intelligent manufacturing systems, power grids, and automotive systems. The ability to test system behaviors under realistic conditions and in a safe and controlled environment makes HIL simulation an essential tool for engineers and researchers.

3.1.1 HIL Platform Architecture for Intelligent Electric Drive Systems

Fig. 3.1 depicts an overall architecture for the intelligent electric drive HIL testing platform. The HIL platform primarily consists of four subsystems: an RT simulator, digital control units, a data acquisition system, and a host PC. The RT simulator is the heart of the HIL platform and simulates the physical plants and all environmental factors in real time. The digital control units are microcontrollers (MCUs) or digital signal processors (DSPs), which run testing control algorithms. The data acquisition system is an isolated intelligent monitor that collects and analyzes information from other subsystems without interfering with their operations. Finally, the host PC is the overall system manager, supervising the entire HIL platform.

OPAL-RT OP5700 RT Simulator

The developed HIL testing platform adopts the OPAL-RT OP5700 as its RT simulator. OPAL-RT OP5700 simulator is a real-time simulation platform designed for power electronics and power systems applications. It is a HIL simulator that allows users to test and validate power system designs in a safe and controlled environment, before deploying them in the field. The OP5700 simulator provides high-fidelity, real-time simulation of complex power system models, with a scalable architecture that allows for the testing of large-scale systems. The OP5700 simulator is equipped with powerful hardware and software components, including high-performance processors, FPGA-based real-time simulation engines, and a real-time operating system. The simulator also includes a comprehensive suite of software tools for

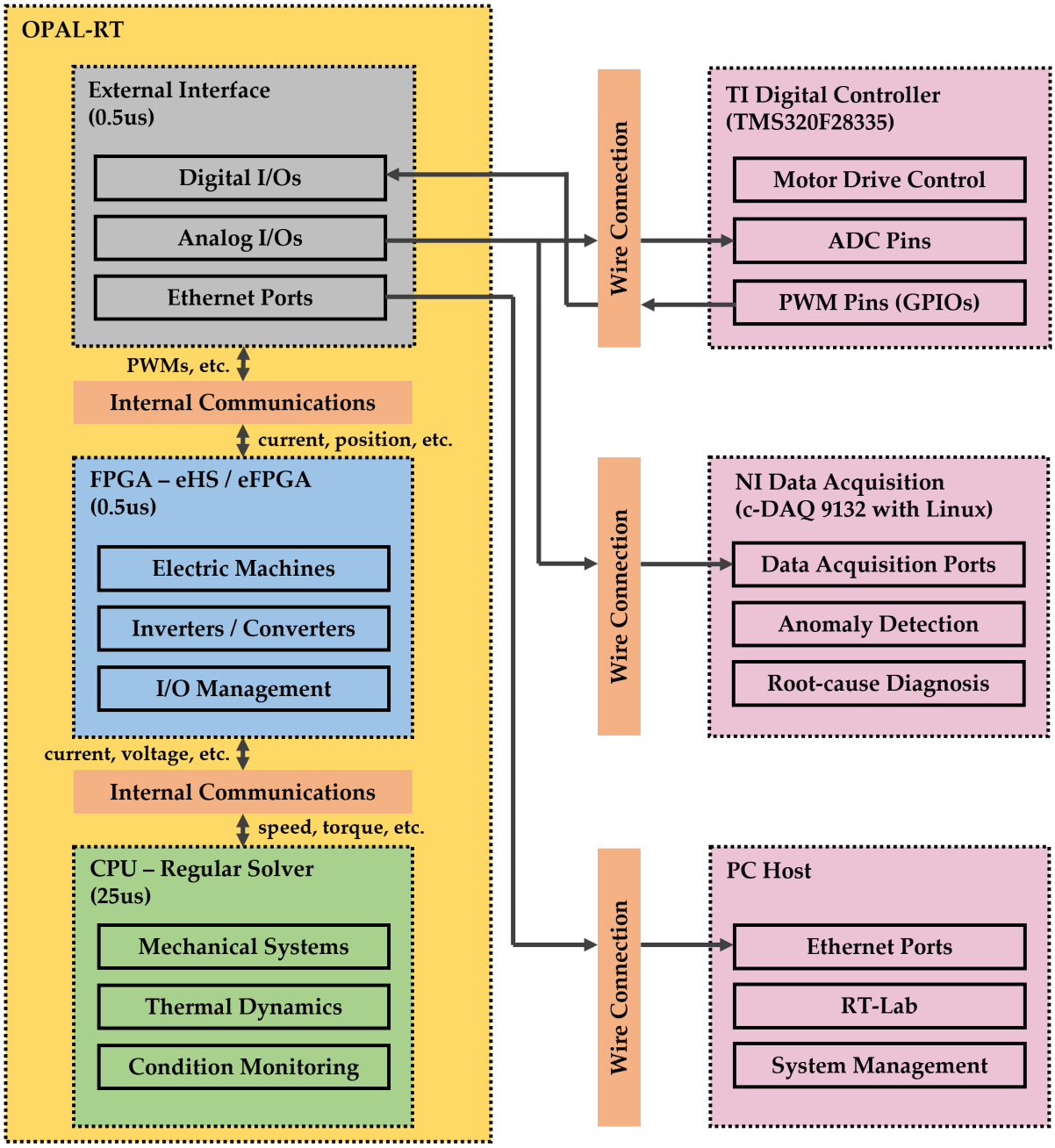


Figure 3.1: Diagram of HIL testing platform architecture for intelligent electric drive systems.

modeling, simulation, and analysis, such as the OPAL-RT Real-Time Simulation (RT-LAB) software, which provides a user-friendly interface for creating and running real-time simulations. One of the key features of the OP5700 simulator is its ability to interface with real-world hardware components, such as

power electronics devices, sensors, and actuators. This allows users to test and validate their designs in a closed-loop configuration, where the simulated system interacts with real-world hardware components. The OP5700 simulator also supports a wide range of communication protocols, including Ethernet, CAN, and serial interfaces, enabling seamless integration with different types of hardware. The OP5700 simulator has been used in a wide range of power electronics and power systems applications, such as renewable energy systems, electric vehicle charging stations, and smart grid systems. It has been used to test and validate complex control strategies, such as the coordination of multiple distributed energy resources, and the integration of renewable energy sources into the power grid. The OP5700 simulator has also been used to test and validate the performance of protection systems and fault detection algorithms, which are critical for ensuring the safety and reliability of power systems.

In the developed HIL testing platform, as shown in fig. 3.1, the OP5700 simulator includes three primary function blocks: CPU block, FPGA block, and external interface block.

The CPU block implements a regular solver with a 25us simulation step time. It simulates slow-dynamic systems and environmental factors, such as mechanical systems, thermal dynamics, and system-level operating commands.

The FPGA block consists of a faster solver with 0.5us simulation step time. It handles the simulation of fast-dynamic systems, such as electric machines, power electronics inverters, and physical fault mechanisms.

The external interface block is integrated with the hardware FPGA units. It manages the information flows at the digital and analog I/Os and the communications between the simulator and external devices.

Texas Instruments (TI) C2000 TMS320F28335 MCUs

The Texas Instruments (TI) C2000 TMS320F28335 is a high-performance microcontroller designed for real-time control applications. It is based on the C28x core, which is a highly optimized digital signal processing (DSP) core designed for high-speed control applications. The TMS320F28335 has a clock speed of up to 150 MHz, and includes a range of peripheral interfaces, such as multiple serial communication ports, analog-to-digital converters, and pulse-width modulation (PWM) outputs. One of the key features of the TMS320F28335 is its ability to perform real-time control tasks, such as motor control, power electronics, and digital power conversion. It includes advanced control algorithms, such as proportional-integral-derivative (PID) controllers and vector control algorithms, which are optimized for high-speed control applications. The TMS320F28335 also includes hardware acceleration for fast Fourier transforms (FFTs) and other signal processing algorithms, which enable efficient implementation of advanced control algorithms. The TMS320F28335 is also designed to operate in harsh industrial environments, with a wide operating temperature range and high resistance to electromagnetic interference (EMI) and electrostatic discharge (ESD). It also includes advanced features for system-level protection, such as a memory protection unit and a watchdog timer, which enhance system reliability and safety. In addition to its real-time control capabilities, the TMS320F28335 includes a range of software development tools and resources, such as the Code Composer Studio integrated development environment (IDE) and the TMS320x2833x

device family reference manual. These tools and resources enable efficient software development and debugging, and provide users with access to a large community of TI developers and engineers.

In the developed HIL testing platform, TMS320F28335 MCUs run the control algorithms for different motor drive units. It receives the simulated three-phase line currents and rotor position signals from the OP5700 simulator's analog output ports. It then converts the analog signals into digital values using the onboard analog-to-digital conversion (ADC) blocks. Meanwhile, it outputs the PWM control commands from the implemented control algorithms and sends them to the digital input ports of the OP5700 simulator. Various emulated cyber-attack scenarios are also implemented through TMS320F28335 for cyber-physical safety and security studies.

National Instruments (NI) cDAQ-9132 Data Acquisition System

The National Instruments (NI) cDAQ (CompactDAQ) is a modular data acquisition system designed for use in a variety of applications, including control and monitoring, test and measurement, and automation. The cDAQ system consists of a chassis and a range of I/O modules that can be customized to meet the specific needs of each application. The cDAQ chassis provides a platform for connecting multiple I/O modules, and includes built-in features for synchronization, triggering, and signal conditioning. One of the key features of the cDAQ system is its flexibility and modularity. Users can choose from a range of I/O modules, such as analog input, analog output, digital input, digital output, and counter/timer modules, to build a customized data acquisition system that meets their specific requirements. The cDAQ system supports a range of signal types, including voltage, current, temperature, strain, and frequency, and includes built-in signal conditioning and filtering to ensure accurate measurements. Another important feature of the cDAQ system is its ease of use and integration with other NI software and hardware. The cDAQ system is compatible with NI LabVIEW, a graphical programming environment for data acquisition, analysis, and visualization, as well as other NI software tools, such as NI-DAQmx and NI Measurement and Automation Explorer (MAX). The cDAQ system also includes a range of drivers and APIs for integrating with other third-party software and hardware. The cDAQ system is designed for use in a wide range of applications, including research and development, testing and validation, and process control. It is well-suited for applications that require high-speed, high-accuracy measurements, as well as those that require synchronized acquisition and control of multiple channels. The cDAQ system is widely used in industries such as automotive, aerospace, industrial automation, and life sciences.

The developed HIL testing platform adopts a specific cDAQ-9132 chassis, which has a powerful integrated CPU running a Linux real-time operation system (RTOS). As shown in Fig. 3.1, the cDAQ system is the core for the isolated system monitor. It could test and validate various developed detection and diagnostic algorithms, including model-based and data-driven methods, due to the powerful CPU.

3.1.2 Example HIL Platform: Dual-Motor Electric Vehicle Powertrain

Last decade has witnessed the fast progress of electrification of modern vehicles. Nevertheless, the accompanying concerns about immature on-board electrical and intelligent technologies remains a large obstacle

for modern techniques to completely replacing the traditional vehicular technologies. Recently, an increasing amount of work has been devoted to cyber-physical security research for modern electric vehicles. In 2010, Koscher et al. experimentally evaluate the cyber-physical security issues on a modern automobile and demonstrate the fragility of the underlying system structure.(Koscher et al., 2010) In addition, the impact of cyber attacks on electric drives are studied in (Yang et al., 2019a) and (Yang et al., 2019b); and in (Broggi et al., 2012; Cui et al., 2012; Dan & Sandberg, 2010; Kwon et al., 2013; Martini et al., 2015; Mitchell & Chen, 2013; Vuković & Dán, 2013; D. Wu et al., 2016), both model-based and data-driven approaches for cyber-physical security are studied as well. Furthermore, due to the complexity of the system configuration and the high cost of manufacturing real-world testbed, real-time and hardware-in-the-loop (HIL) simulation have been widely adopted in the research and development for advanced on-board systems such as electrified powertrain and on-board communication networks. For example, (Tabbache et al., 2012) proposed a simple HIL simulation system for the induction motor-based powertrain coupled to a DC machine-based load torque emulator taking into account the electric vehicle mechanics and aerodynamics; (Hongyu et al., 2015) adopted HIL simulation for developing advanced control strategy for a pure electric vehicle; (Araújo et al., 2016) constructed the battery model and corresponding battery management system (BMS) in the real-time simulation environment; (Awadallah et al., 2018) used HIL to test an electric propulsion system used in a mild hybrid electric vehicle powertrain.

Due to the fact that most of the relevant research about electric vehicle powertrain still focuses on developing advanced and optimal control strategies, most of the simulation platforms only discuss one subsystem in detail, such as battery management systems (BMSs) or electric drive systems (EDSs). For the purpose of reducing computation burden, many nonlinear features have been neglected in designing specific systems. When considering the problems of cyber-physical security, more comprehensive simulation platform is needed in the following aspects:

1. Cyber-physical attacks are considered as random behaviors that could occur anywhere in the powertrain;

2. The impact of cyber-physical attacks will not be limited to one specific system, but the entire powertrain;

3. System nonlinearity is one of the features the cyber-physical attackers could exploit to cause more drastic damage;

4. Cyber-physical attacks could have both short and long term impacts on the system.

Therefore, to accurately reflect the impact of different cyber-physical attacks and evaluate the system reliability and security, an advanced simulation model with the ability of real-time simulation and details like tire-road interactions, high-frequency switching of power electronics and electrical-thermal coupling within the electric drive systems is required. This section describes a HIL platform of a dual-motor based electric vehicle powertrain based on the architecture in previous sections. Such a platform includes all the details shown in Fig. 3.2. More specifically, it includes multiple physical domains of an electric vehicle powertrain: electric drive systems, mechanical transmission system, battery system, and vehicle control

units. Meanwhile, an advanced energy consumption monitor that calculates the system energy and power information is constructed. In addition, other critical details like tire-road interactions and road and aerodynamic friction information are also taken into considerations.

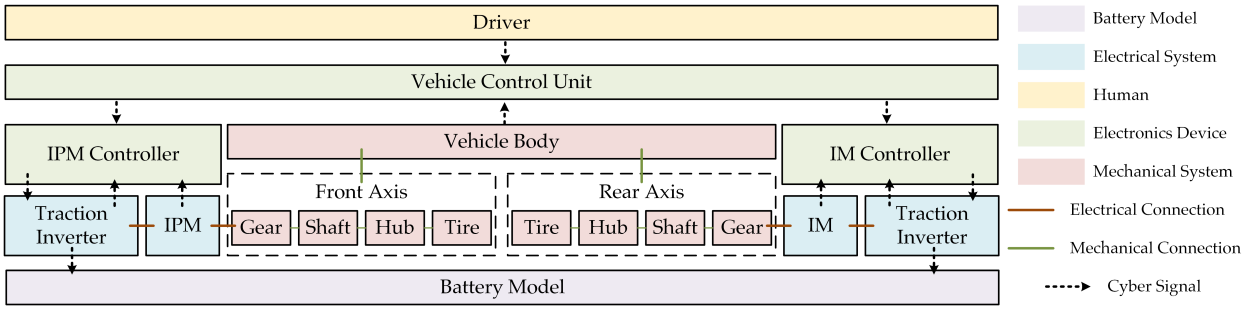


Figure 3.2: System diagram of dual-motor based electric vehicle powertrain.

Dual-Motor Electric Vehicle Powertrain

In this platform, we adopted a widely used powertrain structure, the dual-motor based powertrain. The system diagram is shown in Fig. 3.2, which includes two different motors driving the front and real axis, respectively. As shown in Fig. 3.2, the front wheel is driven by an interior permanent magnet synchronous machine (IPM), and the rear wheel is driven by an induction machine (IM). Each electric machine has its own controller and traction inverter to control the power flows between machines and the battery. In addition, the vehicle control unit (VCU) gathers information from the driver's demands and the vehicle body, such as the vehicle speed, and then provides the torque reference to each of the motor controllers. The detailed models of each subsystem will be described in the following sections.

IPM Based EDS Model

Due to the high power density and smooth torque production, IPM has been widely used as the traction motor of electric vehicles. Fig. 3.4 shows the configuration of the IPM based EDS. Referring to the torque command received from the vehicle control unit (VCU) and the feedback signals gathered from sensors, a current controller is adopted alongside the maximum torque per ampere (MTPA) algorithm to generate the pulse width modulation (PWM) signals, which control the traction inverter fed by a DC power supply and then drive the IPM. In this study, for the purpose of achieving real-time simulation, the eFPGA solver and eHS solver of OPAL-RT is adopted to solve the machine and power

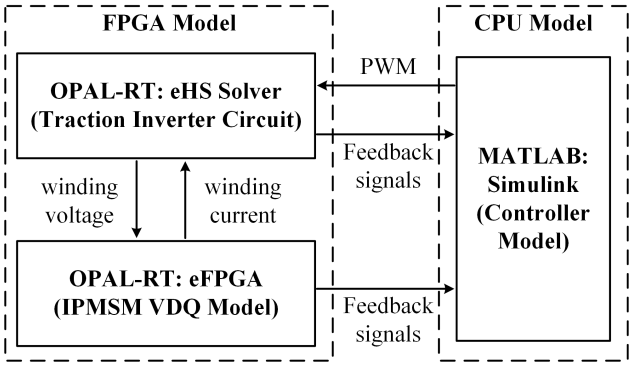


Figure 3.3: The configuration of the EDS model.

electronics models and to realize the simulation with a time step of $0.5\mu s$; and the controller model is constructed by MATLAB Simulink, which is run in the CPU with a time step of $25\mu s$.

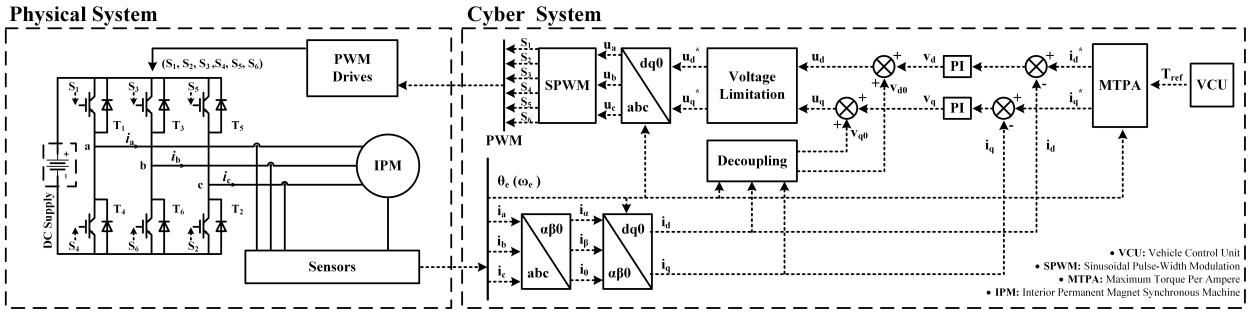


Figure 3.4: Schematic diagram of the IPM drive including both cyber system (control algorithms) and physical system (hardware).

The simulation configuration is shown in Fig. 3.3. The reason why adopting eFPGA and eHS solver is that the ability of parallel computing of FPGA enables simulation with time step as low as $0.25\mu s$, which makes it possible for power electronics models with 10kHz to 200kHz switching frequency to run in real-time. As shown in Fig. 3.3, the electric circuits of traction inverter is solved in eHS, and the IPM is modeled by its Variable D-Q model (VDQ) from eFPGA library, which is derived by the following equations under direct-quadrature-zero (DQZ) reference frame:

$$u_d = R_s i_d + L_d(i_d, i_q)\frac{di_d}{dt} - \omega_e L_q(i_d, i_q) i_q \tag{3.1}$$

$$u_q = R_s i_q + L_q(i_d, i_q)\frac{di_q}{dt} + \omega_e (L_d(i_d, i_q) i_d + \phi_{pm}) \tag{3.2}$$

$$T_e = \frac{3}{2} p(\phi_{pm} i_q + (L_d(i_d, i_q) - L_q(i_d, i_q)) i_d i_q) \tag{3.3}$$

where $u_d,\ u_q,\ i_d,\ i_q,\ \phi_{pm}$ are the d-axis and q-axis voltage and current vectors, and flux linkage generated by permanent magnet, respectively; p is the number of pole pairs; and $L_d(i_d, i_q),\ L_q(i_d, i_q)$ are the nonlinear inductance of d- and q- axis acquired by 2-D look-up tables.

Meanwhile, the MTPA algorithm is derived from the following optimization problem:

$$\min \quad i_d^2 + i_q^2 \tag{3.4}$$

$$s.t. \quad \begin{cases} \frac{3}{2} p(\phi_{pm} i_q + (L_d(i_d, i_q) - L_q(i_d, i_q)) i_d i_q) = Te \\ \sqrt{i_d^2 + i_q^2} \le I_{am} \\ \sqrt{u_d^2 + u_q^2} \le U_{am} \end{cases} \tag{3.5}$$

where T_e is the anticipating torque and $u_d,\ u_q$ are obtained by Eq. (3.1) and Eq. (3.2).

Vehicle Plant Model

To reflect the mechanical characteristics as detailed as possible, the vehicle plant model covers shaft stiffness, load distributions between the front and rear axis, the tire-road interactions, rolling resistance, gradient resistance, and aerodynamic resistance. The shaft model is modeled as:

$$T = K_s \cdot \phi \qquad \text{(rotational spring)} \tag{3.6}$$

$$T = K_d \cdot \Omega \qquad \text{(rotational damper)} \tag{3.7}$$

where T is the torque applied to the shaft, K_s, K_d are the damping coefficients, and ϕ, Ω are the rotational angle and speed, respectively. The tire-road interaction is derived from the tire magic formula. First of all, the dimensions of vehicle plant and the force conditions of the tire is shown in Fig. 3.5; and the traction force, the tire slip κ and nominal load distribution $F_{z-front}$, F_{z-rear} are defined:

$$\kappa = \frac{|r_w \cdot \Omega - v_x|}{|v_x|} \tag{3.8}$$

$$F_{z-front} = F_z \cdot (1 - \frac{l_f}{l}) - \frac{h}{l} \cdot F_t \tag{3.9}$$

$$F_{z-rear} = F_z \cdot \frac{l_f}{l} + \frac{h}{l} \cdot F_t \tag{3.10}$$

where F_z is the total nominal load, and F_t is the vehicle total traction force. Then, according to the magic formula of tire-road interactions, the horizontal traction force F_x is defined:

$$F_x = F_{z-j} \cdot D \cdot \sin(C \cdot \arctan(B\kappa - E \cdot (B\kappa - \arctan(B\kappa)))) \tag{3.11}$$

where B, C, D, E are the constant coefficients of magic formula, which is dependent on the load conditions, and F_{z-j} is the nominal load distribution defined in Eq. (3.9), Eq. (3.10), (j denotes the front or the rear wheel).

In addition, the rolling resistance, gradient resistance and the aerodynamic resistance are also considered as the road resistance force in this model, which are derived from:

$$F_g = k_f \cdot \cos\alpha \qquad \text{(rolling resistance)} \tag{3.12}$$

$$F_r = m \cdot g \cdot \sin\alpha \qquad \text{(gradient resistance)} \tag{3.13}$$

$$F_w = \frac{1}{2} C_d \cdot A \cdot \rho \cdot v^2 \qquad \text{(aerodynamic resistance)} \tag{3.14}$$

where k_f is the rolling resist coefficient, m is the mass of the vehicle, g is the acceleration of gravity; α is the road gradient, and C_d, A, ρ, v are the drag coefficient, reference area, air density and vehicle speed, respectively.

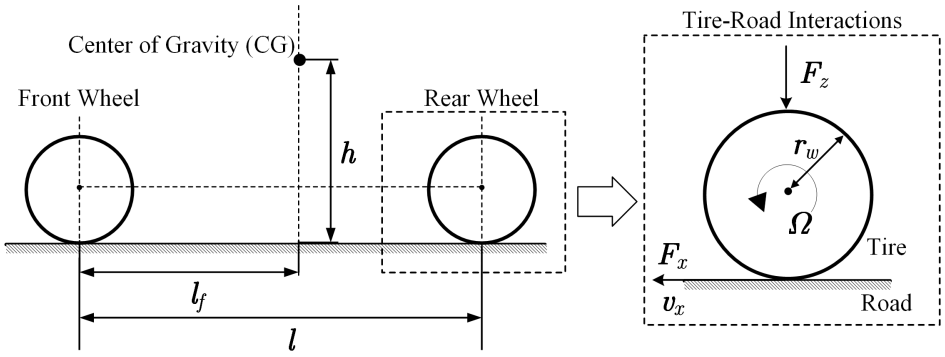


Figure 3.5: Vehicle dimensions and the force conditions.

Energy Consumption Monitor

In the application of automobiles, energy efficiency is always a crucial topic. Therefore, in this platform, we construct an energy consumption monitor to calculate the instant energy information for further analysis. The monitor is divided into four parts: motor loss, inverter loss, mechanical loss, and battery loss. Each part will be elaborated more as follows:

Motor Loss: The power loss in electric machines includes winding copper loss, iron core loss, solid loss, and mechanical loss. Due to the nonlinearity of electric machines and the difficulties of power calculations, we use ANSYS to generate the efficiency map of the electric machines by conducting Finite Element Analysis. Fig. 3.6 shows the resulted IPM efficiency map.

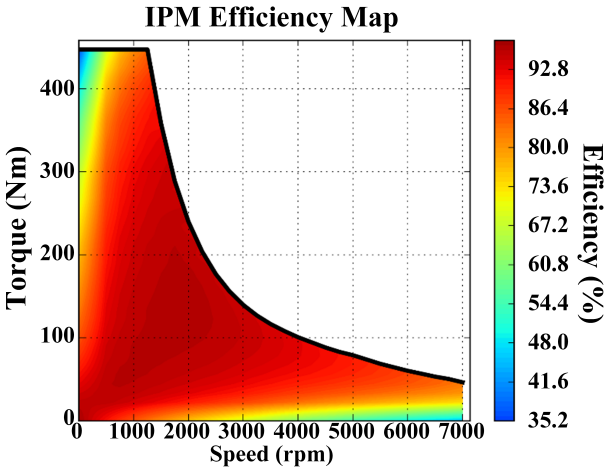


Figure 3.6: Efficiency map of IPM generated by Finite Element Analysis.

Traction Inverter Loss: Recently, with the pervasive adoption of power electronics devices in automobile applications, the energy efficiency of the traction inverters is getting increasing attention. In this platform, we use the electro-thermal model in (H. Ye et al., 2014; J. Ye et al., 2017) to calculate the instant power loss of the traction motor. The detailed calculation procedures are shown in Fig. 3.7.

Mechanical Loss: Due to the fact that the tire-road interactions are taking into consideration, the mechanical loss becomes another important factor of vehicle energy consumption. Meanwhile, as the mechanical transmission system is highly nonlinear, and the calculation of individual components is

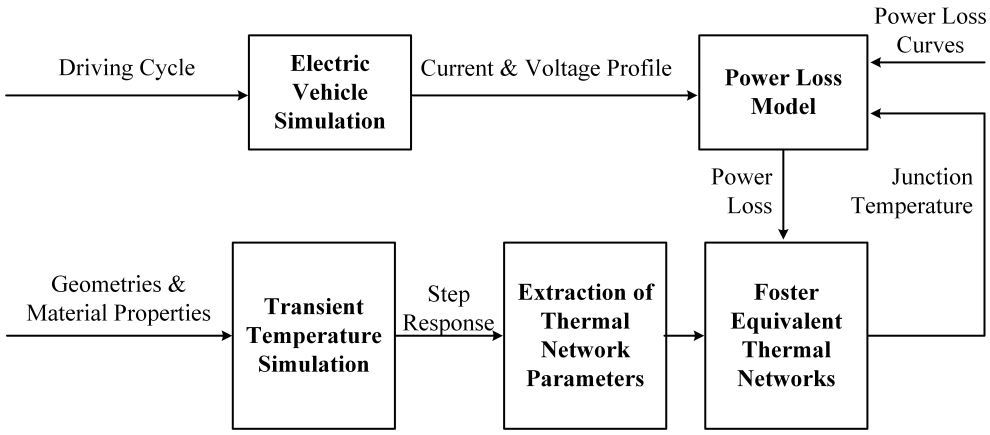


Figure 3.7: Instant power loss calculation procedures using electro-thermal model.

difficult, we focus on the input and output of the entire mechanical system, instead. The input power is from the traction motors, which is calculated by Eq. (3.15), and the output power is calculated from the horizontal vehicle movement, which is derived by Eq. (3.16). T_{ipm}, T_{im} are electromagnetic torque generated by IPM and IM; $\omega_{ipm}, \omega_{im}$ are the rotational speed of IPM and IM; F_trac is the estimated total traction force; and v is the vehicle speed.

$$p_{in} = T_{ipm} \cdot \omega_{ipm} + T_{im} \cdot \omega_{im} \quad \text{(input power)} \tag{3.15}$$

$$p_{out} = F_{trac} \cdot v \quad \text{(output power)} \tag{3.16}$$

Battery Loss: The battery resistance model is adopted to calculate the energy and power information during charging and discharging. The energy relationship is derived from Eq. (3.17) and Eq. (3.18),where p_s, p_1 are the battery input power and output power; U_s, I are the battery voltage and current; and R is the internal resistance.

$$p_s = U_s \cdot I = p_1 + I^2 \cdot R \quad \text{(power equation)} \tag{3.17}$$

$$I = \frac{U_s - \sqrt{U_s^2 - 4p_1R}}{2R} \quad \text{(current solution)} \tag{3.18}$$

With the power information, the Sate of Charge (SOC) could be calculated by Eq. (3.19)

$$SOC = \frac{\int_0^t p_s dt}{C_s} \tag{3.19}$$

where C_s is the total capacity of the battery.

Simulation Results

The simulation testbed is shown in Fig. 3.1 and validated by the New European Driving Cycle (NEDC). The simulation results will be discussed in detailed from the following aspects: mechanical performance, electrical performance, energy monitor, and attack case study.

Mechanical Performance: Fig. 3.8 shows the profiles of vehicle speed, front wheel torque, and front wheel slip under NEDC. As shown in Fig. 3.8, the entire system is operating stably, the slip is within 0.01, which is conform to the reality with a good road condition.

Electrical Performance: Fig. 3.9 shows the profiles of IPM three phase current, IPM torque, and IPM traction inverter loss under piece-wise NEDC. As shown in Fig. 3.9, the torque ripple is less than 5%. The total harmonic distortion of the three phase current is not calculated due to the fact that the fundamental frequency is varying with respect to the vehicle speed. Nevertheless, the current distortion is minor from Fig. 3.9. Meanwhile, it could also be seen that the power loss of traction inverter is fluctuating around certain values.

Energy Monitor: Fig. 3.10 shows the profiles of vehicle speed, total efficiency, and subsystem efficiency under piece-wise NEDC. It could be seen that the traction inverter is operating with the efficiency over 95% and both traction motors are operating with the efficiency around 90%. In addition, the total traction motor drive's efficiency is around 87%, which conform to the real world data.

Attack Case Study: Fig. 3.11 shows the profiles of IPM three phase current, IPM d-axis current, and IPM q-axis current when the IPM drive is under a malicious attack. In this case, the attack happens at 108.00s, which disables the current controller of IPM by injecting false values into the phase A current sensor feedback signals. As shown in Fig. 3.11, when the attack happens, the three phase current will be seriously distorted and the d- and q- axis current will also be deviated from their normal values.

According to the simulation results, the strength of this platform could be summarized as follows:

1. Features from multiple physical domains of the electric vehicle powertrain could be reflected;
2. Energy consumption information could be calculated for further analysis;
3. Real-time simulation could realize the simulation of long-period driving cycles;
4. Physics-based features and patterns could be generated for cyber attack scenarios.

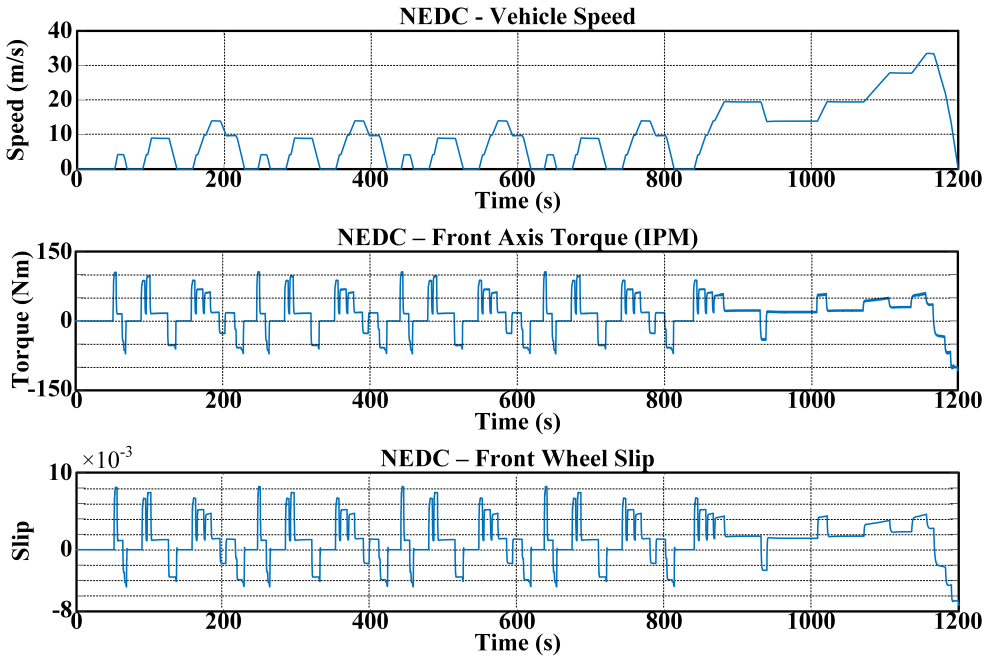


Figure 3.8: Simulation results: profiles of vehicle speed, front wheel torque, and front wheel slip under NEDC.

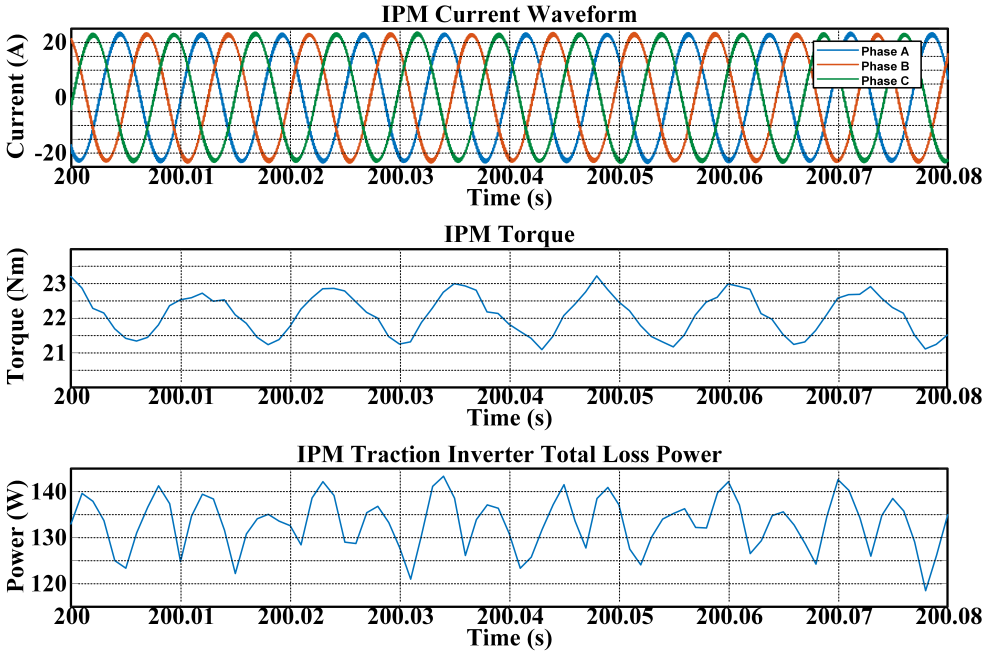


Figure 3.9: Simulation results: profiles of IPM three phase current, IPM torque, and IPM traction inverter loss under piece-wise NEDC.

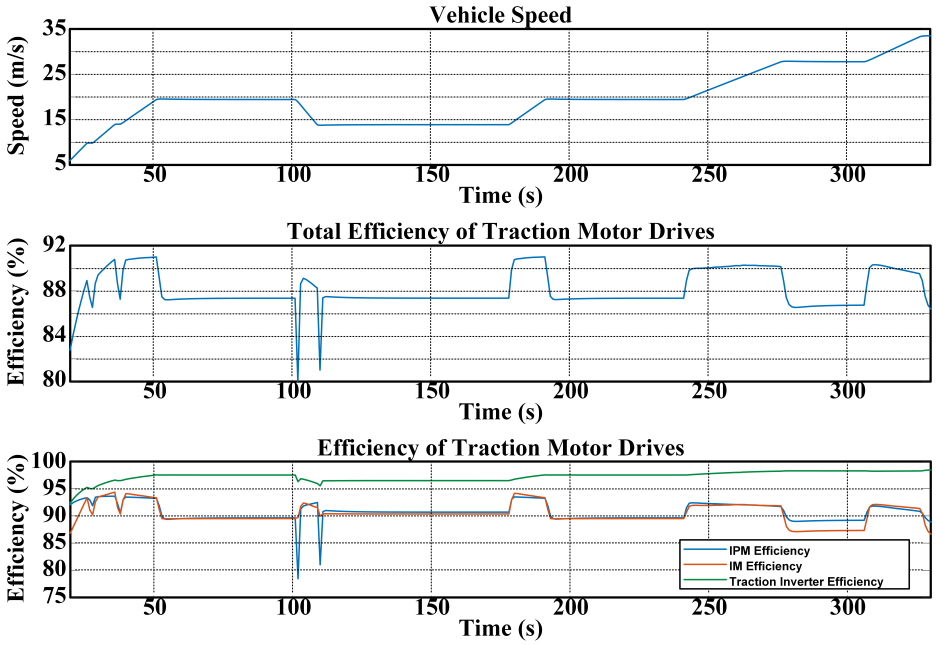


Figure 3.10: Simulation results: profiles of vehicle speed, total efficiency, and subsystem efficiency under piece-wise NEDC.

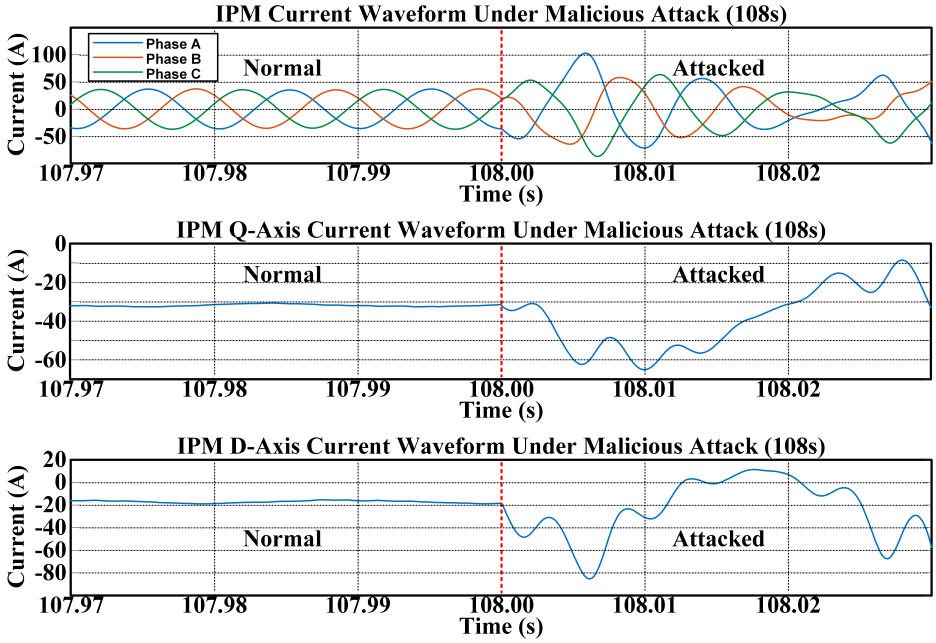


Figure 3.11: Simulation results: profiles of IPM three phase current, IPM d-axis current, and IPM q-axis current when the IPM drive is under malicious attack.

3.2 Hardware Experiment Platform

While the HIL simulation platform is a low-cost, risk-free, and effective tool for analyzing system behaviors, a scaled real-world hardware experiment platform is still desirable due to the following reasons.

1. Hardware experiment platform could verify the authenticity of the simulated environment factors in the HIL simulation platform.
2. Hardware experiment platform could assess the impacts of additional factors not considered in the HIL simulation platform.

Therefore, this section described a developed real-world hardware experiment platfor for intelligent electric drive system.

3.2.1 System Architecture

Fig. 3.12 shows a diagram and a photo of the developed hardware experiment platform. This platform includes four electric machine drives sharing the same DC power supply. Four digital control units control the four electric machine drive units, respectively. Furthermore, each control unit has a host PC connected through lab networks, emulating the operating networks in real-world applications. The NI cDAQ-9132, with similar configurations to the HIL simulation platform, also forms the isolated monitoring system. The goal of this hardware experiment platform is:

1. Emulate real-world operating environment and generate authentic data sets with various scenarios.
2. Verify the results from the HIL simulation platform.
3. Test and validate developed detection and diagnostic algorithms in a real-world environment.
4. Demonstrate the developed isolated monitoring solution.

Electric Drive Units

The developed hardware experiment platform includes two induction machines (IMs) and two permanent magnet synchronous machines (PMSMs). All electric drive units have a Texas Instruments (TI) C2000 TMS320F28335 microcontroller as the digital control unit.

The TMS320F28335 is a high-performance microcontroller designed for real-time control applications. It is based on the C28x core, which is a highly optimized digital signal processing (DSP) core designed for high-speed control applications. The TMS320F28335 has a clock speed of up to 150 MHz, and includes a range of peripheral interfaces, such as multiple serial communication ports, analog-to-digital converters, and pulse-width modulation (PWM) outputs. One of the key features of the TMS320F28335 is its ability to perform real-time control tasks, such as motor control, power electronics, and digital power conversion.

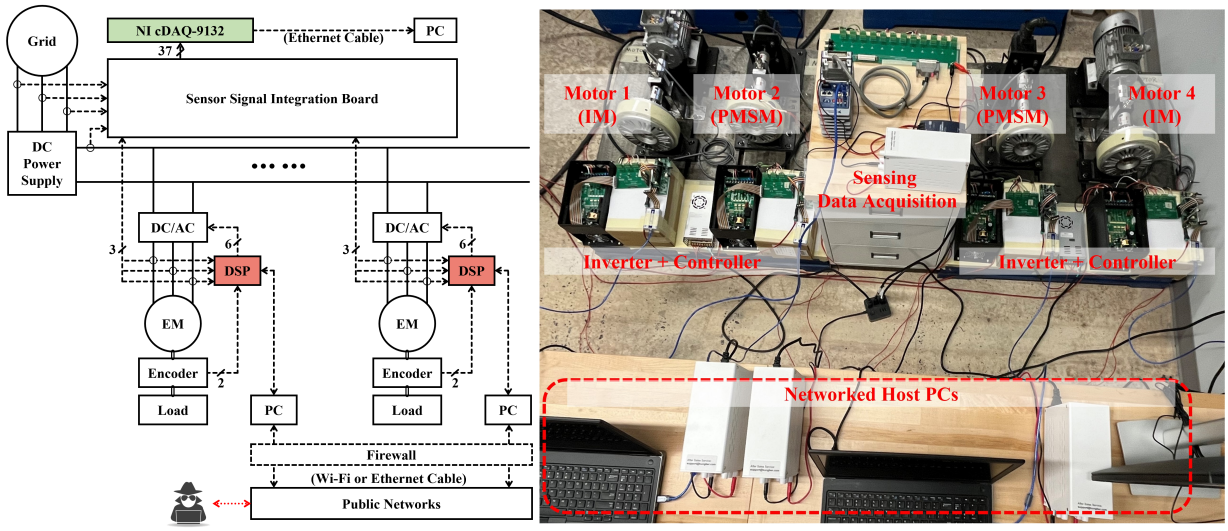


Figure 3.12: Diagram and photo of the developed hardware experiment platform.

It includes advanced control algorithms, such as proportional-integral-derivative (PID) controllers and vector control algorithms, which are optimized for high-speed control applications. The TMS320F28335 also includes hardware acceleration for fast Fourier transforms (FFTs) and other signal processing algorithms, which enable efficient implementation of advanced control algorithms. The TMS320F28335 is designed to operate in harsh industrial environments, with a wide operating temperature range and high resistance to electromagnetic interference (EMI) and electrostatic discharge (ESD). It also includes advanced features for system-level protection, such as a memory protection unit and a watchdog timer, which enhance system reliability and safety. In addition to its real-time control capabilities, the TMS320F28335 includes a range of software development tools and resources, such as the Code Composer Studio integrated development environment (IDE) and the TMS320x2833x device family reference manual. These tools and resources enable efficient software development and debugging, and provide users with access to a large community of TI developers and engineers.

In addition, these four electric drive units adopt field-oriented control (FOC) strategy to track the rotating speed commands. FOC is a popular control strategy for both induction machines and permanent magnet synchronous machines. FOC involves controlling the currents in the stator windings of the machine to produce a rotating magnetic field that follows the rotor field. This approach provides a high level of control over the machine's torque and speed, making it a suitable choice for many industrial applications.

In an induction machine, the stator and rotor magnetic fields are not directly coupled, and the rotor field lags behind the stator field. To apply FOC to an induction machine, the stator currents must be transformed from the stationary reference frame (abc) to the rotor reference frame (d-q). This transformation involves calculating the rotor flux angle and using it to transform the stator currents. In FOC for

induction machines, the goal is to control the stator currents such that the rotor flux angle follows the stator flux angle. The rotor flux angle can be estimated using a sensorless approach, such as the sliding mode observer. The control system then adjusts the stator current magnitude and phase to track the desired rotor flux angle. The stator current phase is adjusted using a proportional-integral (PI) controller, which compares the actual and desired rotor flux angles and generates a current command that is fed into the PI controller. The magnitude of the stator current is adjusted using a current regulator, which uses the PI controller output to generate a voltage command. This voltage command is then fed into a pulse-width modulation (PWM) converter to produce the desired stator current. The FOC approach provides precise control over the induction machine's torque and speed, even under varying loads and operating conditions. It also allows for smooth starting and stopping of the machine and can reduce energy consumption by optimizing the machine's efficiency.

In a permanent magnet synchronous machine (PMSM), the rotor magnetic field is produced by permanent magnets, and the stator field is created by the stator currents. FOC for PMSMs is similar to that for induction machines, but the transformation from the stationary reference frame to the rotor reference frame is simpler since the rotor field is already known. In FOC for PMSMs, the stator current is controlled such that the rotor flux angle follows the stator flux angle. The control system includes a PI controller that compares the actual and desired rotor flux angles and generates a current command that is fed into the PI controller. The magnitude of the stator current is adjusted using a current regulator, which uses the PI controller output to generate a voltage command. This voltage command is then fed into a PWM converter to produce the desired stator current. FOC for PMSMs provides excellent control over the machine's torque and speed, and is widely used in applications such as electric vehicles and industrial automation. It allows for high-precision control over the machine's output, even under varying loads and operating conditions, and can improve the machine's efficiency by reducing losses.

Table 3.1: Specifications of the PMSMs in the developed hardware experiment platform.

Rated Power	1.5 kW	Stator Resistance	0.4050 Ω
Rated Current	8.2 A	Stator Inductance	0.0024 H
DC Bus Voltage	200 V	Magnet Flux Linkage	0.0599 Wb
Rated Frequency	250 Hz	Number of Pole Pairs	5
Control Frequency	10 kHz	Motor Inertia	3.10e-4 kgm^2

Table 3.2: Specifications of the IMs in the developed hardware experiment platform.

Rated Power	1.5 kW	Stator Resistance	1.85 Ω
Rated Current	3.4 A	Stator Inductance	0.1084 mH
DC Bus Voltage	200 V	Rotor Resistance	1.98 Ω
Rated Frequency	150 Hz	Rotor Inductance	0.1116 H
Control Frequency	10 kHz	Number of Pole Pairs	3

Fig. 3.13 shows the control diagram of FOC for PMSMs and IMs, and Table 3.1 and Table 3.2 show some detail parameters for PMSMs and IMs.

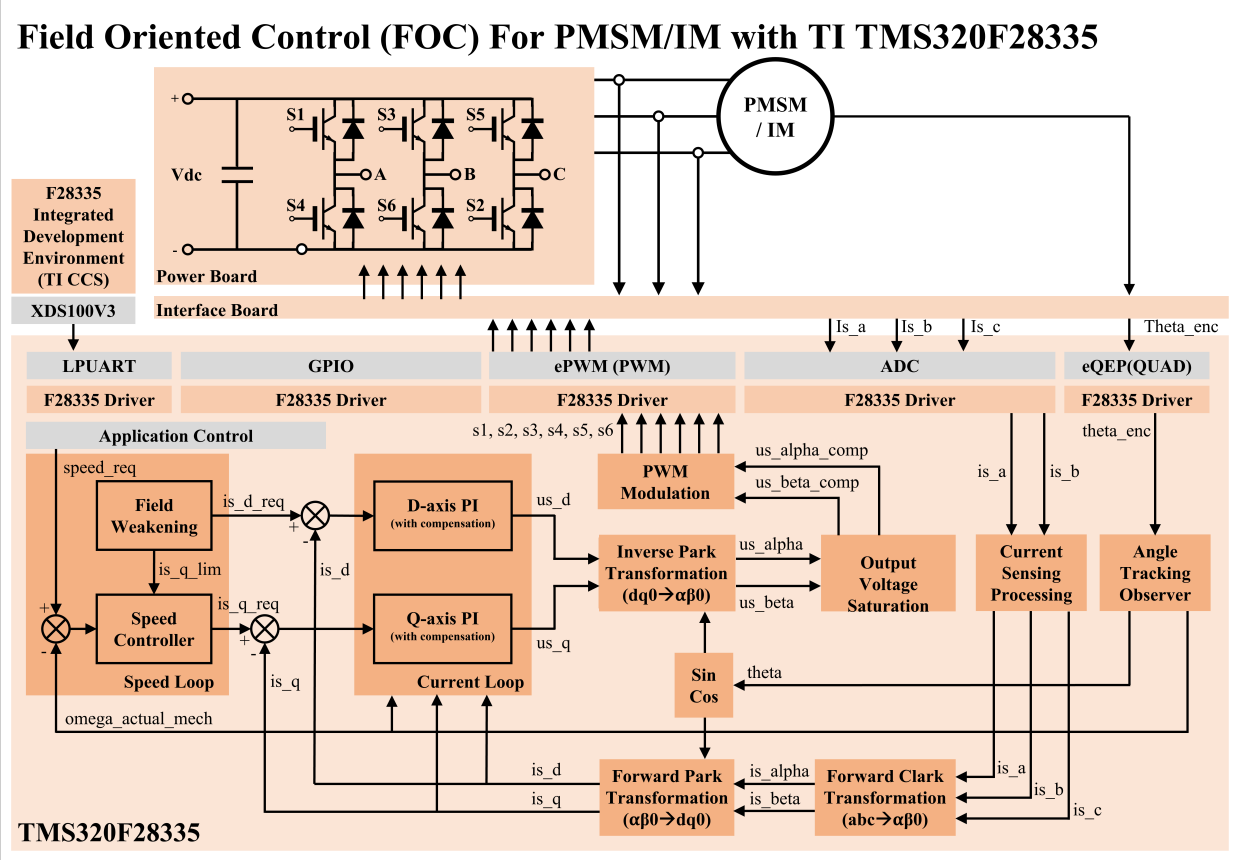


Figure 3.13: Controller diagram for field-oriented control (FOC) for PMSM and IM.

Isolated Monitoring System

The isolated monitoring system is similar to the one in the HIL simulation platform, which is discussed in previous sections. The difference in the hardware experiment platform is that it has an extra sensor integration board, as shown in Fig. 3.12. This sensor integration board integrates the signals from remote current sensors at each motor, DC bus, and PCC. This integration board then uses a DB37 connection to route these signals to the NI cDAQ-9132 chassis. This setup allows the cDAQ to select desired signals with high flexibility.

Cyber-attack Scenarios

Realizing cyber-attacks in real-world hardware platforms is always challenging due to the trade-off between the fidelity and controllability of the attack scenarios. An ideal attack scenario should be one of the real-world attacks from historical studies. However, cyber-attacks are highly unpredictable, and the impacts on

the physical systems are even more challenging to manage. Therefore, the developed hardware experiment platform adopts fully controllable false data injection attack (FDI) scenarios. The emulated attacks are pre-defined and embedded in the TMS320F28335 MCUs with some triggers. These triggers are backdoors for various FDI attacks. Therefore, the attack policies could be fully controllable, and the impacts on physical systems could be manageable. The hacker then could trigger one of the pre-defined backdoors, and the impacts could be analyzed through the acquired data sets from the cDAQ systems.

3.2.2 Experiment Results of a Sample Cyber-Attack Scenario

This section shows some experiment results from a sample false data injection attacks (FDIAs) targeting one of the PMSM electric drive unit.

False data injection attacks targeting closed-loop control systems are a type of cyber-attack that aims to manipulate the output of a control system by injecting malicious data into its input. This type of attack is particularly dangerous because it can cause the system to behave in unexpected and potentially damaging ways. In a closed-loop control system, the output of the system is fed back into the input, where it is used to adjust the control action. An attacker can exploit this feedback loop to inject false data that modifies the output of the system in a way that is detrimental to the control objectives.

One common approach for FDIAs in closed-loop control systems is to manipulate the sensor measurements that are used as feedback signals. For example, an attacker can inject false measurements that cause the control system to make incorrect decisions, such as increasing the output of a system beyond its safe limits. In some cases, the attacker may also modify the control commands that are sent to the actuators, which can cause the system to behave in unexpected ways. These attacks can be challenging to detect because the injected data may appear to be valid sensor readings, making it difficult for the control system to distinguish between normal and malicious inputs.

FDIAs targeting closed-loop control systems are a growing concern in many critical infrastructure systems, including power grids, water distribution systems, and transportation networks. These systems rely heavily on closed-loop control systems to maintain their safe and reliable operation, making them attractive targets for attackers seeking to disrupt their function. To mitigate the risk of these attacks, researchers are developing new detection and mitigation techniques that can identify and block malicious inputs in real-time. These techniques often involve advanced machine learning algorithms that can detect patterns in the sensor data that are indicative of an attack, allowing the system to take corrective actions before any damage is done.

Fig. 3.14 to Fig. 3.17 shows the target PMSM three-phase line currents and DC bus line current of a sample FDIA targeting the phase A ADC offset variable. The bias injected is 0.1, which represents around 1A bias in the machine feedbacks.

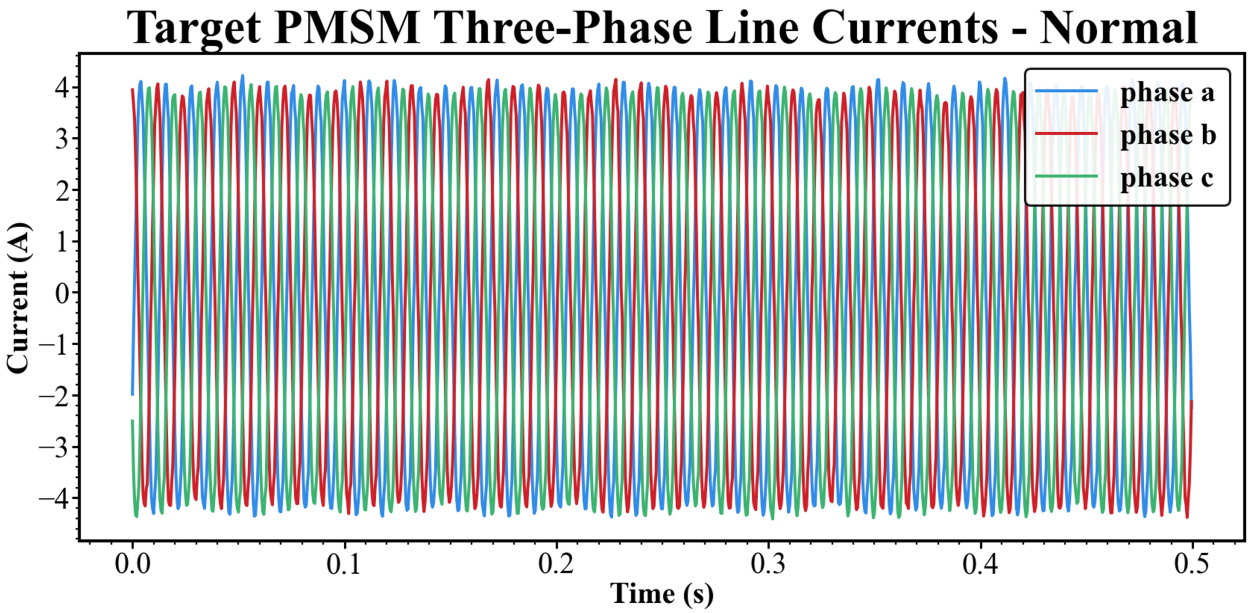


Figure 3.14: Recored raw waveforms of target PMSM three-phase line currents (Normal Condition).

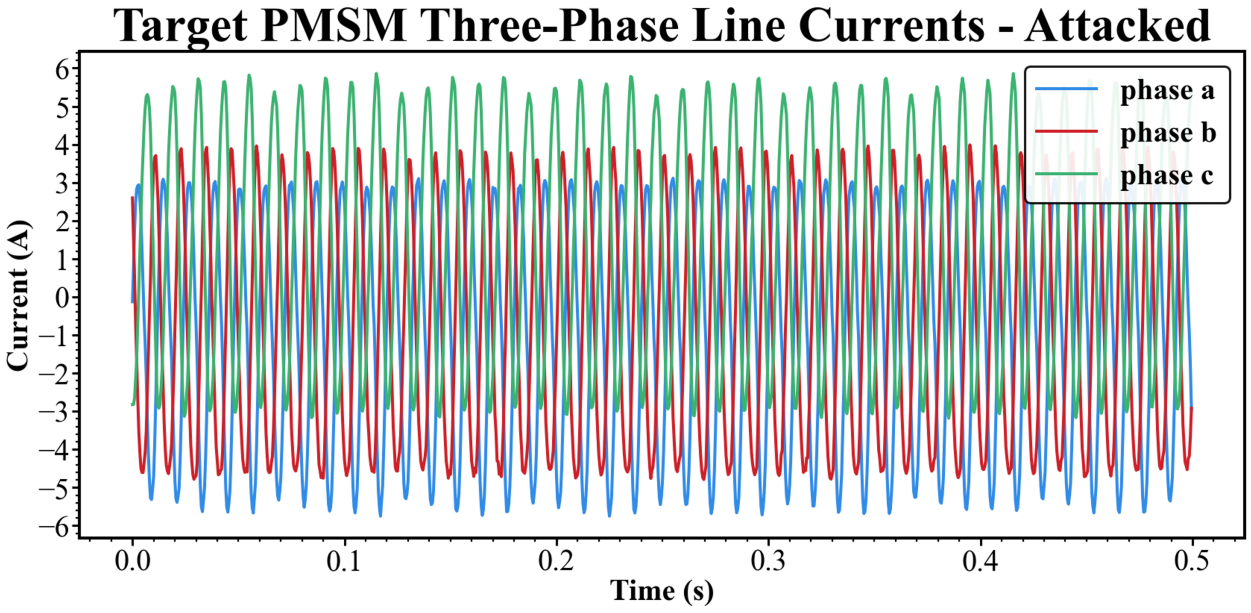


Figure 3.15: Recored raw waveforms of target PMSM three-phase line currents (Under Attack).

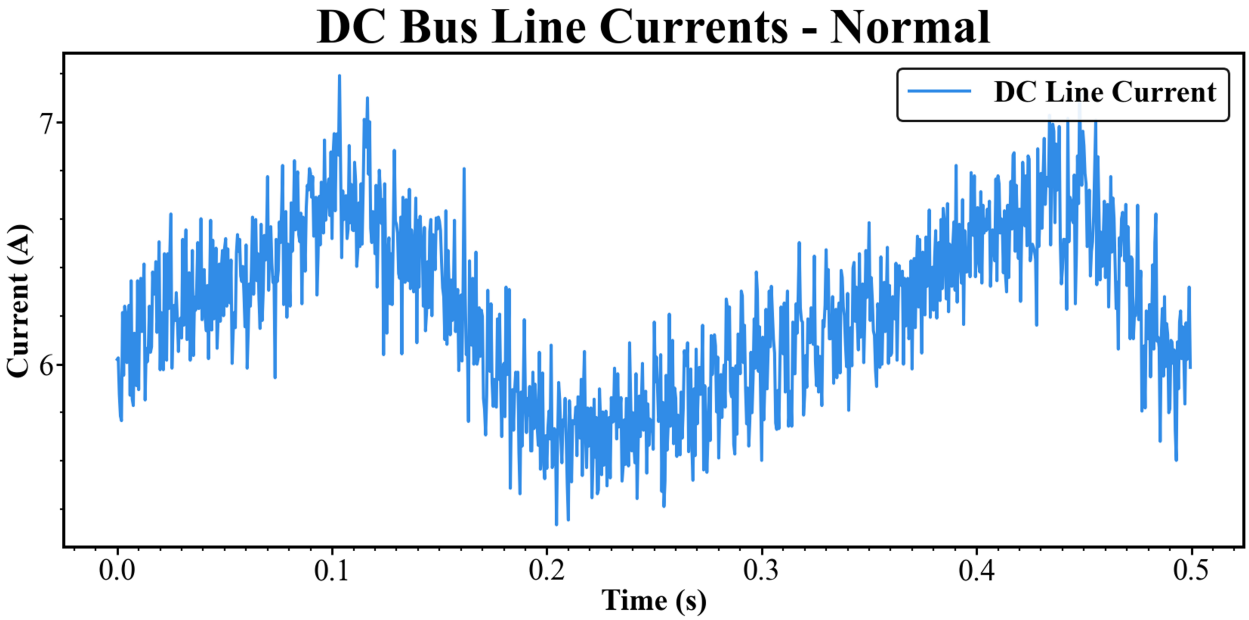


Figure 3.16: Recored raw waveforms of DC bus line current (Normal Condition).

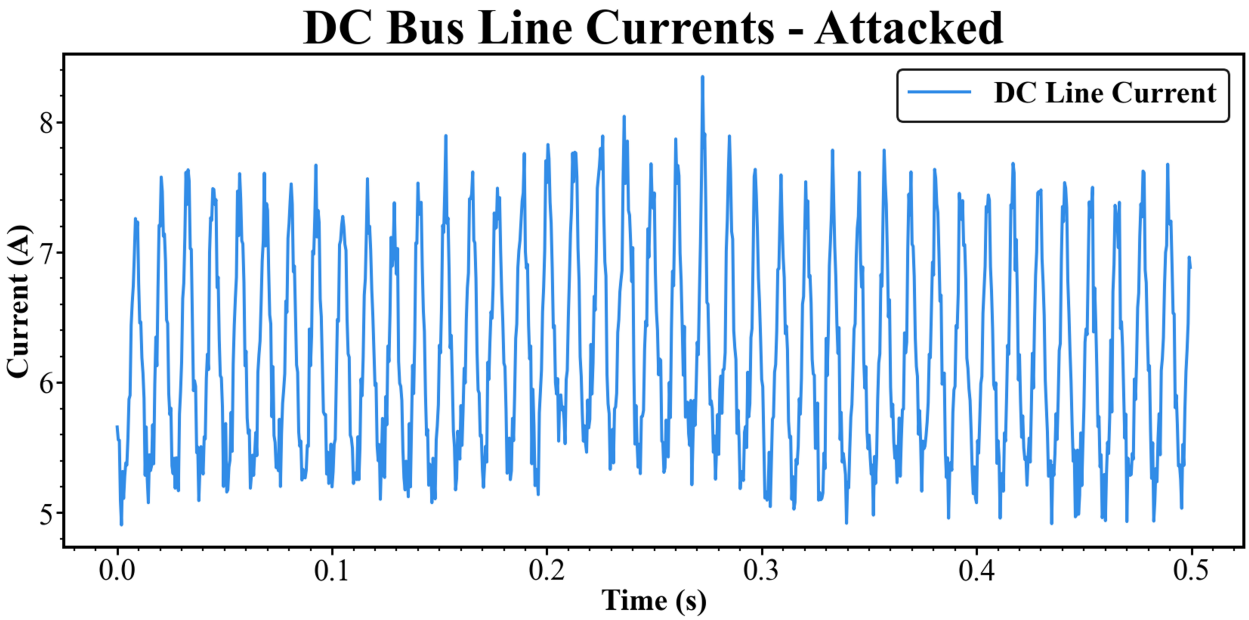


Figure 3.17: Recored raw waveforms of DC bus line current (Under Attack).

Chapter 4

Cyber-Physical Security and Safety Monitoring Algorithms

4.1 Fast Detection for Cyber Threats in Electric Vehicle Traction Motor Drives using Time-Domain Features

In recent years, researchers have witnessed the drastic development of electric vehicles (EVs). Meanwhile, due to the benefits of the modern electric platforms, the vehicle onboard communication networks and Vehicle-to-X (V2X) techniques are also developed to adapt to the market requirement of advanced vehicle functionalities and performances. Therefore, large amounts of electronic units are deployed to the vehicle networks to realize modern vehicle technologies, such as online powertrain optimizations and advanced driver assistance systems (ADAS). According to (Charette, 2009), even for a premium-class automobile in 2009, the vehicle contains approximately 100 million lines of codes executed on 70-100 electronic control units (ECUs). While this progress has driven significant advancements in efficiency, functionality, and safety, it has also brought cyber threats, including electronic device failures and malicious cyber attacks. These threats could cause severe consequences to the drivers, passengers, and surrounding traffic systems. In 2010, Koscher et al. experimentally evaluated the cyber-physical security issues on a modern automobile and demonstrated the fragility of the underlying system structure (Koscher et al., 2010). They demonstrated that an attacker who can infiltrate virtually any ECUs could completely circumvent a broad array of safety-critical systems. Over a range of experiments, both in the lab and in the road test, they demonstrated the ability to control a wide range of automotive functions adversarially while ignoring the driver's requirements, for instance, disabling the brakes, power interruption, and other safety-critical functions. Recently, the IEEE power electronics society (PELS) launched a cyber-physical-security initiative to address cyber networks' reliability and security issues in power electronics systems. Furthermore, according to Forbes, more than 150 cybersecurity incidents reported in 2019 targeting the automotive industry. It states that the first vehicle-targeted hack happened in 2002, in which the hackers reprogram powertrain calibrations of Audi, Porsche, and Ford for more aggressive performance. Meanwhile, one of the most recent incidents happened in 2020. The hackers exposed security flaws of Ford and Volkswagen range

from remotely exposing private customer information to disabling the traction control system (Tengler, 2020).

Many communities have studies cyber-physical security issues in the past decades. For example, In (Han et al., 2014), Han et al. classify the intrusion detection techniques for the cyber-physical system from two different aspects based on the proposed four-layer structure of cyber-physical systems. Vuković et al. address the detection and localization of false data injection attacks against state estimation in distribution power systems based on the evolution of the exchanged data and the convergence properties of the distributed algorithms (Vuković & Dán, 2013). Meanwhile, Cui et al. devote their efforts to enriching the detection solutions from the following perspectives: (1) attacker versus defender dynamics; (2) distributed attack detection and state recovery (Cui et al., 2012). Kwon et al. research the intelligent cyber attacks that can avoid being detected by the current monitoring system in (Kwon et al., 2013); and in (Dan & Sandberg, 2010), Dán et al. study the stealthy false-data attacks against state estimators in power systems. Besides the above work, other techniques and methodologies have proven effective for attack detection, such as the onboard self-detection method in (Bezemskij et al., 2017), the collaborative detection strategies in (Broggi et al., 2012; Martini et al., 2015; Mitchell & Chen, 2013; D. Wu et al., 2016) and the remote offloaded detection techniques in (Loukas et al., 2017).

In addition, there are related research in the aircraft community as well. Baskaya et al. reviewed current fault detection and diagnosis methods using machine learning techniques in (Baskaya et al., 2017). Chen et al. proposed an improved version of fault diagnosis method via convolutional neural networks in (Chen et al., 2019). Imai et al. developed a self-healing avionics mechanism using dynamic data-driven approach in (Imai et al., 2019). And Wu et al. analyzed the cascading failure based on operational process states in (Y. Wu et al., 2020).

Nevertheless, most researches from the above communities cannot be directly applied to the EV traction systems for the following reasons:

1. Considering the fact that vehicles are operating in relatively random scenarios while current research targets like power systems (PS) and common process control systems (PCS) have relatively more stationary operation cycles, current approaches developed for PS and PCS may not be feasible. In (Loukas et al., 2019), Loukas et al. point out that most of existing techniques may not be feasible for modern vehicles due to the dynamic and unique operation characteristics.

2. Due to the limitations of vehicle onboard space and computational resources, a simple and fast detection method is required before triggering advanced root cause diagnosis based on resource-consuming diagnosis process like Current Signature Analysis (CSA) (Choi et al., 2017; Culbert & Rhodes, 2005; Thomson & Fenger, 2001), and redundant control (Giraldo et al., 2018).

3. As cyber threats include malicious cyber attacks from public networks, the trusted information is also limited. Most of the current detection methods use signals from networked ECUs, which could already be modified by those attacks. For example, in (Giraldo et al., 2018), they use the control sequence u_k to calculate the residual-based metrics, but actually, u_k itself is not trustworthy due to potential cyber threats. In this case, if the attacker sends a "healthy" u_k to the monitor

but sends a modified $\hat{u}_k$ to the actuator, the attacks could bypass the existing monitoring systems. Therefore, a cyber attack detector using only trustworthy physical signals is preferred to avoid such issues.

Although the cyber-physical security of electric vehicles has received increasing attention , most of the research still focuses on the vehicle level rather than traction motor drives. For example, in (Guo, Yang, & Ye, 2020; Guo, Yang, Ye, Chen, et al., 2020; Guo & Ye, 2020), the cyber-physical security of the energy management system and steering system for electric vehicles are studied, in which the issues of cybersecurity and system stability are addressed. However, as the focus is on the vehicle system level, a linear vehicle model is used for the analysis. Thus, it does not work effectively for device-level analysis such as for powertrain traction motor drives, as these systems suffer from severe nonlinearity and uncertainty.

In this section, we propose a binary-classifier-based fast detection method for cyber-physical security of traction motor drives using four easy-to-get sensor signals. The general diagram of the proposed method is shown in Fig. 4.1. The proposed method includes two stages: the design stage and the implementation stage. Training data sets are collected from the experiments and simulations in the design stage and then fed to a pre-selected statistical learning model to generate the binary-classifier-based detector. Then, in the implementation stage, the monitors acquire real-time measurements, calculate the instantaneous features, and feed the well-trained binary classifier. During the implementation stage, a majority vote mechanism is also included for better detection performance. The novelty and contributions for the proposed method are summarized as follows:

1. The proposed detection method selects motor current and position signals that are easier to obtain and secure compared with than cyber signals, such as control signals, so these signals and proposed algorithms are considered trustworthy against cyber attacks.

2. The proposed detection method uses the motor current signals in the dqo reference frame to undermine the impacts of the vehicle random driving cycles. The reason is that the operational patterns for motor currents are limited by the motor control algorithms regardless of the vehicle driving cycles. For example, if the traction drives are well controlled by Maximum Torque Per Ampere (MTPA) and proportional-integral (PI) controllers, the normal current features under the dqo reference frame should be restricted to certain boundary established by the control algorithms.

3. The proposed detection method achieves much faster detection compared to traditional current signature analysis through selecting a set of innovative time-domain current features. These time-domain features are selected to be the most vulnerable to a wide range of anomalies, so a shorter time period of observations is needed, largely reducing the computational burden and the time-to-detect.

4. The proposed detection method does not rely on the physical model of motor drive systems compared to traditional residual-based methods that estimate or predict information from the linear model. Thus, strong nonlinearity and uncertainty of motor drive systems can be better addressed to improve detection accuracy and robustness.

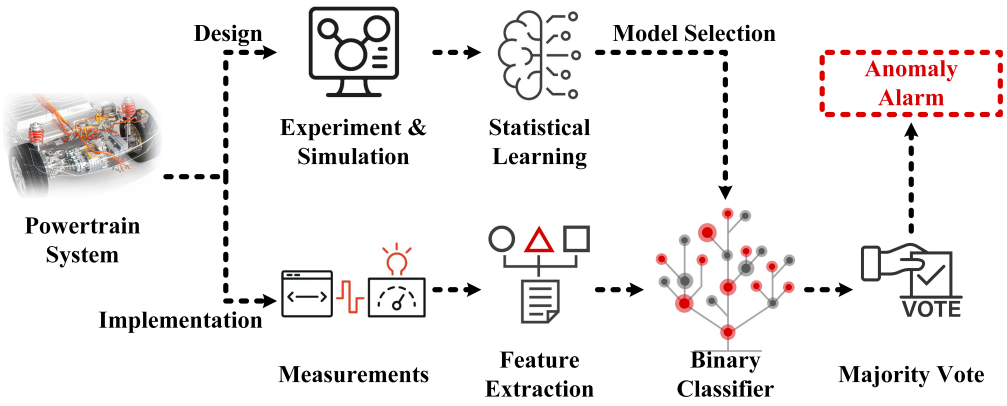


Figure 4.1: General diagram for the proposed detection method.

4.1.1 Model Assumptions and Trusted Signals

The general idea of the proposed method is to establish a boundary between all the extracted physical features using binary classifiers which could distinguish whether or not the system is operating under healthy conditions. In order to better illustrate the proposed approach and to maintain the focus of cyber threats, some assumptions are elaborated in this section. In addition, the trusted signals assumptions will also be elaborated in this section.

Physical Faults and Failures

Generally, the anomalies of a traction motor drive include both physical faults and cyber threats. However, in this section, we only focus on the cyber threats and assume there are no physical faults or failures in the target systems. The reasons for this assumption could be explained as follows. Firstly, the physical faults could be classified into two categories, one is short term faults, and the other one is long term faults. The former ones have a short transient periods between the fault occurrence and the system failure, such as power switch open circuit faults and electric machine short circuit faults. For such faults, the system operation point will suddenly deviate from the normal trajectory; and such faults could also be detected by the proposed method due to its similar characteristics on the proposed features. For example, Fig. 4.2 shows the three-phase current waveform of the IPM motor drive from a machine short circuit fault case study. In this case, a dual-phase partially short circuit fault is simulated after time 53s. We then extract 50 samples of the fault current waveform and test them using our proposed method. The results show that all 50 samples are correctly detected. The other type of faults have relatively longer transient period between the fault occurrence and the system failure, such as inter-turn short circuit faults of electric machines. These kinds of faults tend to evolve slowly after their appearances, and will only cause severe damages after a certain amount of time, so such faults actually provide much larger time windows for the fault detectors than other faults. In addition, it usually requires long period of observations to extract the evolving trends

of such faults, so these kinds of faults are not the main focuses of the proposed fast detection method. Therefore, to maintain the focus of cyber threats, we will assume that there is no physical fault in the target systems in the rest of this section.

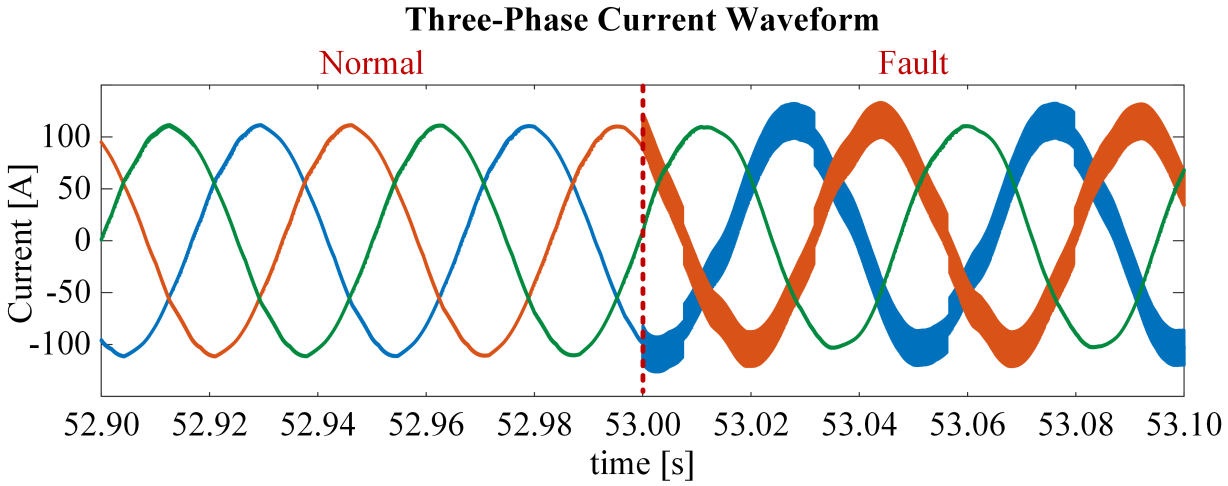


Figure 4.2: Three-phase current waveform of IPM motor drive in a machine short circuit fault case study.

General Control Framework of Traction Motor Drives

In this section, we assume that the traction motor drives have a general control framework shown in Fig. 4.3, which includes networked motor controller (or ECU), traction inverter, electric machine, current sensors and rotor position encoder. More specifically, we will use an interior permanent magnet synchronous machine (IPM) drive for demonstration, whose detailed control diagram is shown in Fig. 4.4. Meanwhile, we also assume that winding current can be directly or indirectly controlled.

System Level Cyber Threats

As shown in Fig. 4.3, the motor controller also communicates with the vehicle on-board networks and exchange information like reference signals and system operation conditions. In this section, we define threats occurring on this communication channel as system level cyber threats and define threats within the motor controller as device level threats. Then, the proposed method only focuses on detecting the device level threats. The reason is that, firstly, as depicted by Fig. 4.5 (Guo, Yang, Ye, Chen, et al., 2020; T. Zhang et al., 2014), in order to detect the system level threats, the detector will need information from other subsystems connected to the vehicle networks, and this target could be achieved by the system level detector. Secondly, as long as the reference signals shown in Fig. 4.3 and Fig. 4.5 do not exceed the safety margins of the traction motor drives, such threats will not cause direct damages to the motor drive systems. Therefore, the proposed method will only focus on the device level attacks, which could cause direct impacts on the motor drive systems.

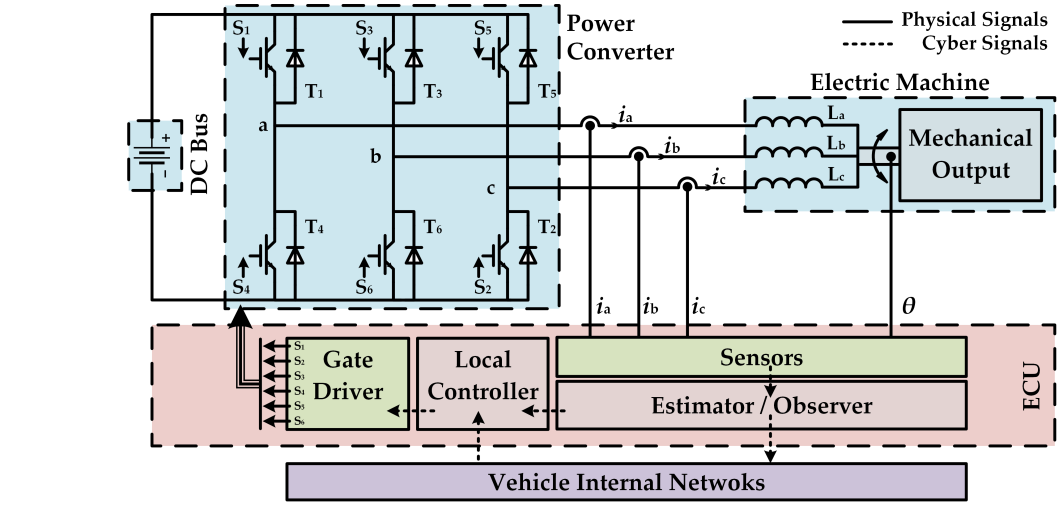


Figure 4.3: General control framework of the traction motor drives.

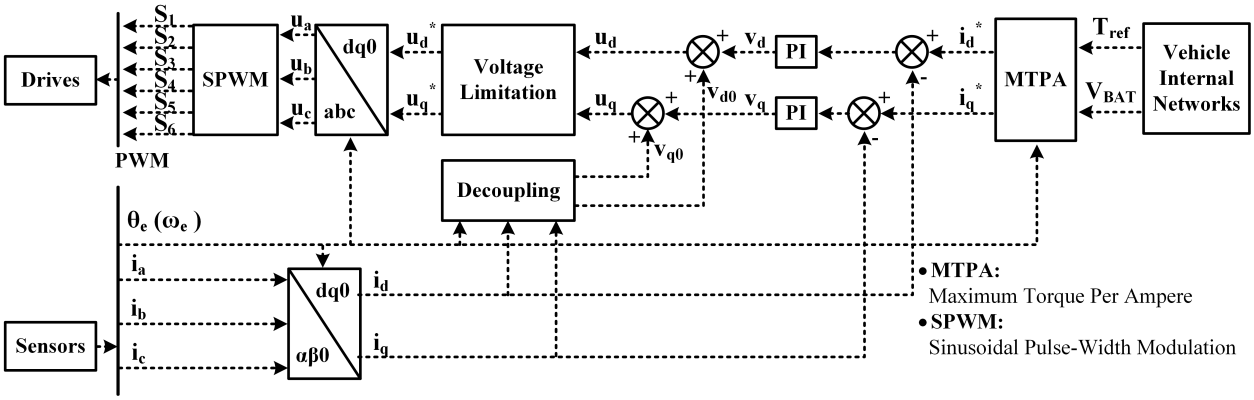


Figure 4.4: Detail control diagram of IPM drive.

Trusted Signals

Whenever dealing with the problems of cyber attacks, the assumption of trusted signals is always one of the most important questions. As shown in Fig. 4.3 and Fig. 4.5, the motor controller is directly connected to the vehicle on-board networks. This indicates that the motor controller is directly exposed to all the cyber threats in the networks. Therefore, all the information and signals in the motor controller are not trustworthy as any one of them may be modified. Fig. 4.6 shows three common cyber attacks targeting on the motor drives, which are:

1. Attacks on the feedback signals, where $y_k \neq \hat{y}_k$;
2. Attacks on the control sequences, where $u_k \neq \hat{u}_k$;

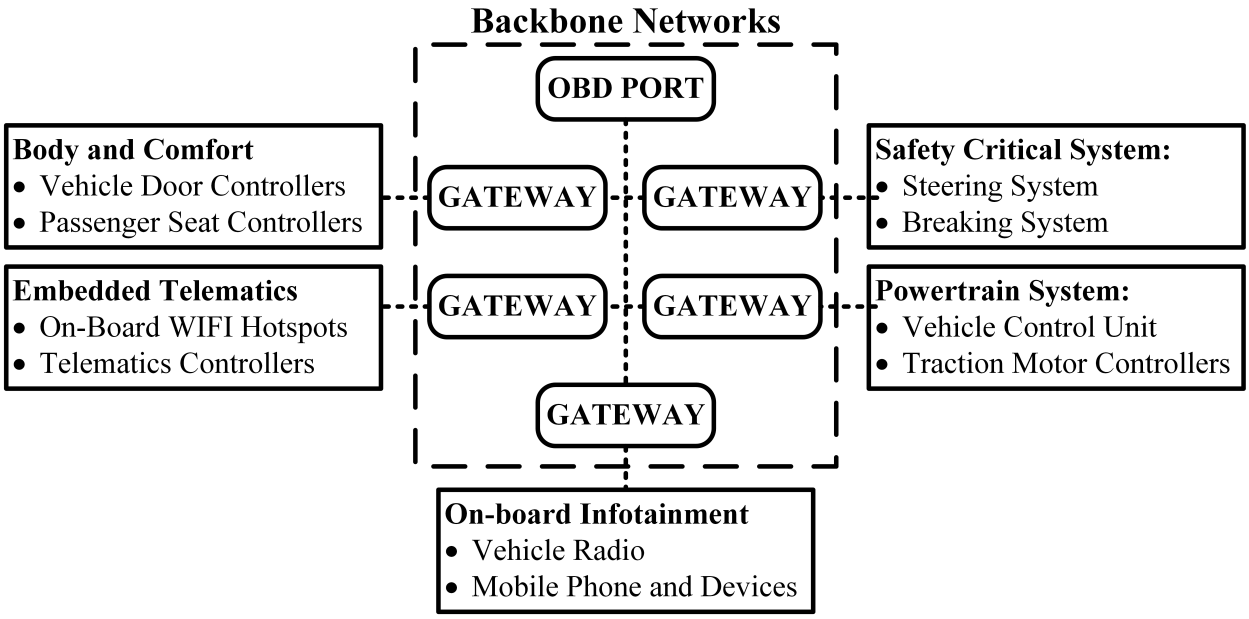


Figure 4.5: General configuration of the EV on-board networks.

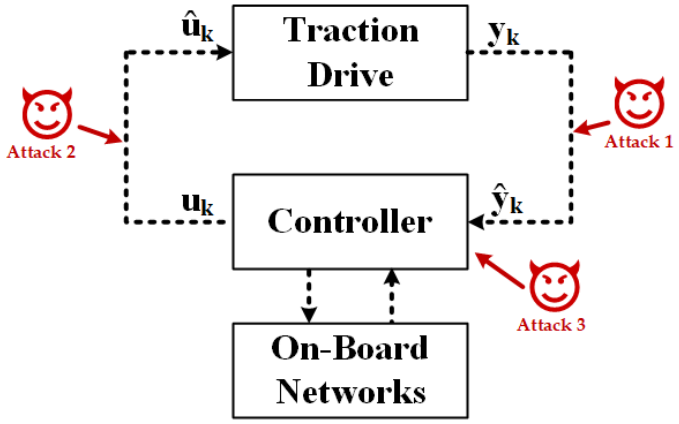


Figure 4.6: Some common attacks targeting on the motor controllers.

3. Attacks on the control parameters.

In this section, we assume four sensor signals are trustworthy, which are three phase current sensor signals $i_a,\ i_b,\ i_c$ and the rotor position encoder signal θ_r. The reasons could be summarized as follows:

1. motor drive sensors are physical components which are not connected to the vehicle networks;
2. extra sensors from a third party, which are completely isolated, could also be easily installed to the system.

For the reasons above, the proposed approach will be based on these four trustworthy signals.

4.1.2 Cyber attack Detection Method Using Binary Classifiers

As shown in Fig. 4.1, the proposed approach is to establish a boundary among all the target system physical features so that the anomalies of the traction motor drive could be detected within a very short time period. The proposed approach could be separated into two stages: the detector design stage and the detector implementation stage.

For the design stage, the general process is described by Algorithm 1, and the major target for this stage is to generate an optimal fast detector for the target system. Details of each steps will be elaborated later in this section.

Algorithm 1 Design Algorithm for Model-free Fast Detector Using Binary Classifiers

1: **Input**: Four sensor data (i_a, i_b, i_c, θ_r) and related condition label.
2: **Output**: Optimal binary classifier based fast detector model.
3: Input data re-sampling using sliding window with length of m and sampling time of t_s;
4: Time domain feature extraction from the re-sampled observations;
5: Model fitting based on the extracted time domain features with different binary classifiers;
6: Optimal Model selection by comparing the k-fold cross validation results of different fitted models;

The second stage is the implementation of the optimal fast detector acquired in the first stage. Fig. 4.7 shows a general diagram of this stage. The general algorithm is shown in Algorithm 2 and each step will be discussed in detail in the following contents.

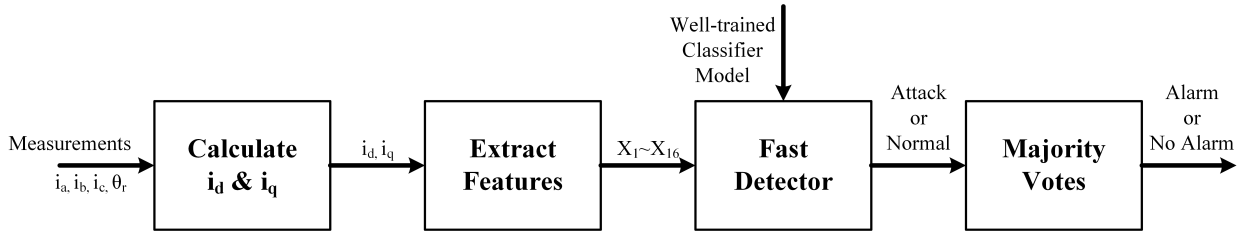


Figure 4.7: General diagram of the implementation stage (Algorithm 2)

Time Domain Feature Extraction

As shown in Algorithm 1 and Algorithm 2, time domain feature extraction plays an important role in both stages, as those features are used as predictors in the binary classifier models. The reason why choosing time domain features is that comparing with frequency domain and time-frequency domain, time domain features could reflect the system instant characteristics with much smaller window size as they do not need to consider the trade off between the frequency resolution and the window size. This point is crucial for the fast detector as large window size will consume a lot of online memory and computational resources.

Algorithm 2 Implementation Algorithm for the Optimal Fast Detector Acquired in the First Stage

Input: Well trained fast detector model and real time measurements of i_a, i_b, i_c, θ_r.
Output: Abnormal detection results.
Generating a sliding window with the same length m and sampling time t_s as in stage 1;
for $k = 0$ **to** $m - 1$ **do**
    read and save i_a, i_b, i_c, θ_r at $t = k * t_s$ to form the initial monitoring window $\mathcal{W}_0$, $t = 0$ is the initial point;
end for
setting the number of voting decision $v = 0$ and total voting capacity as v_{max};
for $k = m, m + 1, m + 2, m + 3, ...$ **do**
    time domain feature extraction from monitoring window $\mathcal{W}_{k-m}$;
    classifying the feature of $\mathcal{W}_{k-m}$ using the input binary classifier model and generate the decision D_{k-m};
    if $v \leq v_{max}$ **then**
        save the decision D_{k-m} as voting candidate V_{k-m};
        $v + +$;
    else
        conducting a majority vote among the voting candidates $V_{k-m-v_{max}}$ to V_{k-m-1};
        if more votes for "threats" **then**
            output "cyber threat alert";
        else
            output "healthy condition";
        end if
        save the decision D_{k-m} as voting candidate V_{k-m};
        delete voting candidate $V_{k-m-v_{max}}$;
    end if
end for

More specifically, the proposed method chooses 16 features from the two traction drives' current signals in the dqo reference frame, i_d and i_q. Such signals are transferred from the original real measurements i_a, i_b, i_c, θ_r using Park Transformation. The expression for the transformation is shown in Eq. (4.1) and Eq. (4.2).

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \frac{2}{3} \cdot \mathbf{P} \cdot \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}. \tag{4.1}$$

$$\mathbf{P} = \begin{bmatrix} \cos(\theta_e) & \cos(\theta_e - \frac{2\pi}{3}) & \cos(\theta_e + \frac{2\pi}{3}) \\ -\sin(\theta_e) & -\sin(\theta_e - \frac{2\pi}{3}) & -\sin(\theta_e + \frac{2\pi}{3}) \end{bmatrix} \tag{4.2}$$

where $\theta_e = \theta_r * polepairs$. In addition, the 16 features extracted from i_d and i_q are developed from the four sample moments (Mean, Variance, Skewness, Kurtosis) of the data within the sliding window, which depict the data distribution characteristics. The detail expression of the four sample moments are shown in Eq. (4.3) to Eq. (4.6).

$$Mean = \mu = \frac{1}{m}\sum_{i=1}^{m}(X_i) \tag{4.3}$$

$$Variance = \sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(X_i - \mu)^2 \tag{4.4}$$

$$Skewness = \mu_3 = \frac{1}{m}\sum_{i=1}^{m}(\frac{X_i - \mu}{\sigma})^3 \tag{4.5}$$

$$Kurtosis = \mu_4 = \frac{1}{m}\sum_{i=1}^{m}(\frac{X_i - \mu}{\sigma})^4 \tag{4.6}$$

where X represents i_d and i_q.

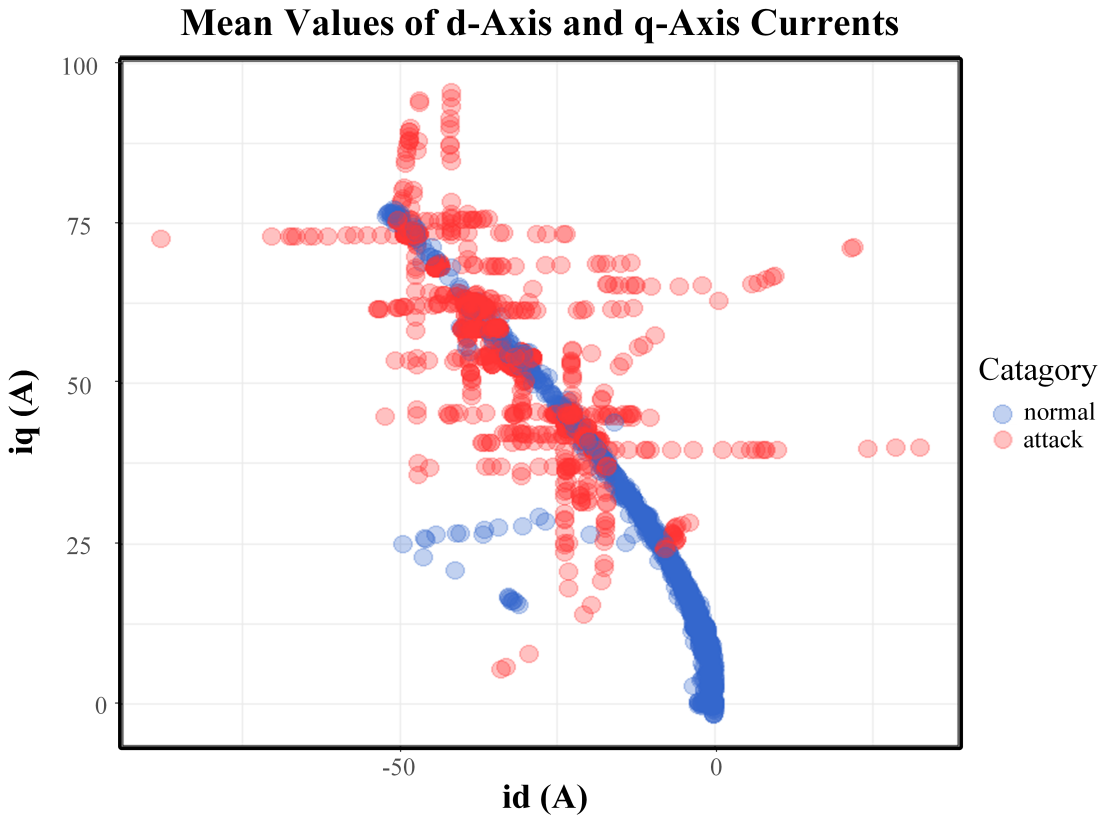


Figure 4.8: A sample of time domain features in dqz reference frame: mean values of d- and q- axis currents from the front wheel drive.

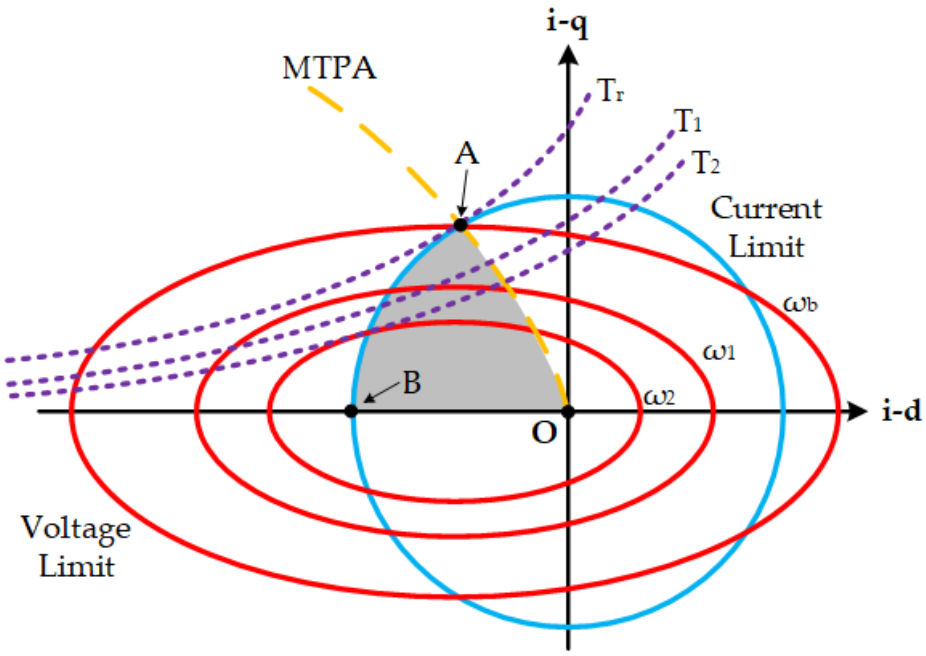


Figure 4.9: Current profiles for IPM drive with MTPA (yellow line).

Generally speaking, these 16 features describe the data distribution of the d-axis and q-axis currents. The reasons for choosing such features could summarized by two aspects: simplicity and interpretability. First, as shown in Eq. (4.3) to Eq. (4.6), the four sample moments are extremely easy to calculate, especially for sliding window, as only two data points will be changed in single step, the first data points in the previous window and the new observation. This largely reduces the online computation requirement and is crucial for the fast detector as discussed in previous sections. Meanwhile, these 16 features could accurately reflect the system operation characteristics from multiple angles. The mean value depicts the instant current level which should alongside the trajectory of the optimized current reference if the system operates in a healthy condition, as the motor drive either directly or indirectly controls the winding currents. For example, in our demonstration, we use an IPM drive with Maximum Torque Per Ampere (MTPA) optimization method, therefore, the mean values of d-axis and q-axis currents should alongside the MTPA current profiles when the IPM drive is operating in a healthy condition. In addition, the variance could reveal the current ripple level, skewness could imply the asymmetry of the current profiles and the kurtosis could examine the extreme values in the sliding window like sudden changes in the system. For example, Fig. 4.8 shows a sample of current mean-value features in dqo frame. It could be seen that each one of the features alone cannot fully distinguish the normal and abnormal conditions due to variation of feature sensitivity in different case scenarios. Therefore, the proposed approach needs to use all 16 features in the binary classifier models. Meanwhile, as discussed in section II, the short term physical faults will have similar characteristics, such as the mean value deviating from the optimized current trajectory, large current ripple reflected by the variance and sudden change of system states implied by skewness and kurtosis. These similarities imply the proposed fast detector could also detect the short term physical faults.

Random Forest Classifier: RF classifier is a modified decision tree based classifier, which operates by constructing a multitude of decision trees at training and outputting the dominant class among all the classes generated from each individual decision trees. RF output model contains the node information and its splitting criteria of all the decision trees in the forest, thus the model computational size depends on the number of nodes and trees in the forest. Therefore, in order to further reduce the consumption of computational resources, it is necessary to simplify the final model by adjusting the predictors with respect to the mean decrease Gini Index of each predictors.

K-Nearest-Neighbor Classifier: KNN classifier is a non-parametric method, which classifies the new observation by a plurality vote of its neighbors in the feature space, and assigns the new observation to the most common class among its k nearest neighbors. KNN is a common classifier with good prediction accuracy, however, due to its non-parametric nature, the KNN classifier tend to occupy a lot more online computational resources, especially when the training data size is large. The reason why KNN classifier is still chosen as the candidate classifier is that for some contexts when the boundary for healthy operation conditions is clear, such as drives with limited operation trajectories, adopting KNN could largely reduce the simulation and experiment cases.

Support Vector Machine Classifier: SVM model is a representation of the training data set as support vectors (or points) in the feature space, which map the training data set to the separate classes dividing by a clear gap; and new observations are then mapped into the same space and predicted to belong to the class based on the side of the gap on which they fall. SVM is a robust classifier with benefits of low consumption of online computational resources as it only need to store the support vectors that map the gap between the two classes.

AdaBoost Classification Trees: AdaBoost (Adaptive Boosting) is a boosting algorithm in machine learning. Improving week learners and creating an aggregated model to improve model accuracy is a key concept of boosting algorithms. A weak learner is defined as the one with poor performance or slightly better than a random guess classifier. AdaBoost classification trees improve those classifiers by increasing their weights and gets their votes to create the final combined model.

Generally speaking, these four classifier candidates represents typical characteristic of the commonly used binary classifiers: RF and AdaBoost methods represent the rule-based classification tree algorithms, KNN represents the non-parametric classifier, and SVM represents the support vector based classifier. The reason for choosing multiple classifier candidates is that the detection results depend on the time domain feature characteristics, such as the optimized current trajectory. Therefore, in order to overcome such variations among different traction drive systems, it is necessary to choose multiple classifier candidates and select the model with optimal performance for stage 2 implementation. For the proposed approach, we adopt k-fold cross validation to estimate the detector performance.

Majority Vote Mechanism

As shown in Algorithm 2, a majority vote mechanism is adopted before the final decision. This is because there are many uncertainties in real world traction motor drives, such as the road conditions and temperature variations. Such external uncertain factors may cause a lot of false alerts, therefore, the majority vote

mechanism could effectively reduce the number of false alerts. In addition, the number of voters v_{max} is chosen with respect to the trade off between the time to detect and the number of false alarms. When v_{max} is large, the time for detection will increase as it requires certain amount of votes to determine the system has cyber threats, but there will be more redundancy for the external uncertain factors. In this section, v_{max} is set to 6.

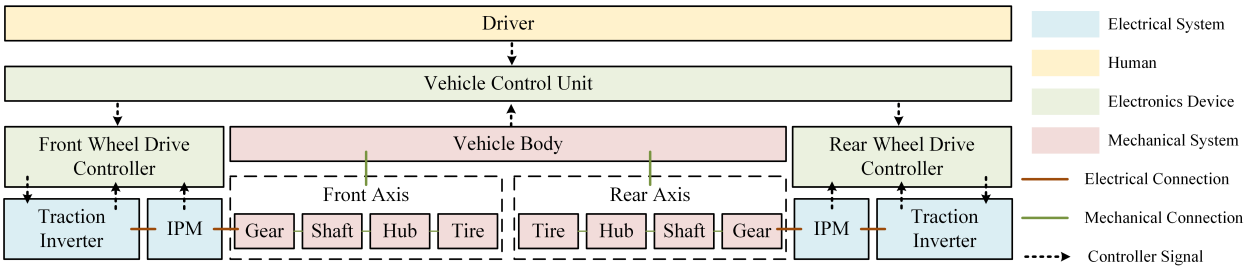


Figure 4.10: General diagram of the powertrain model used in the HIL real-time simulation.

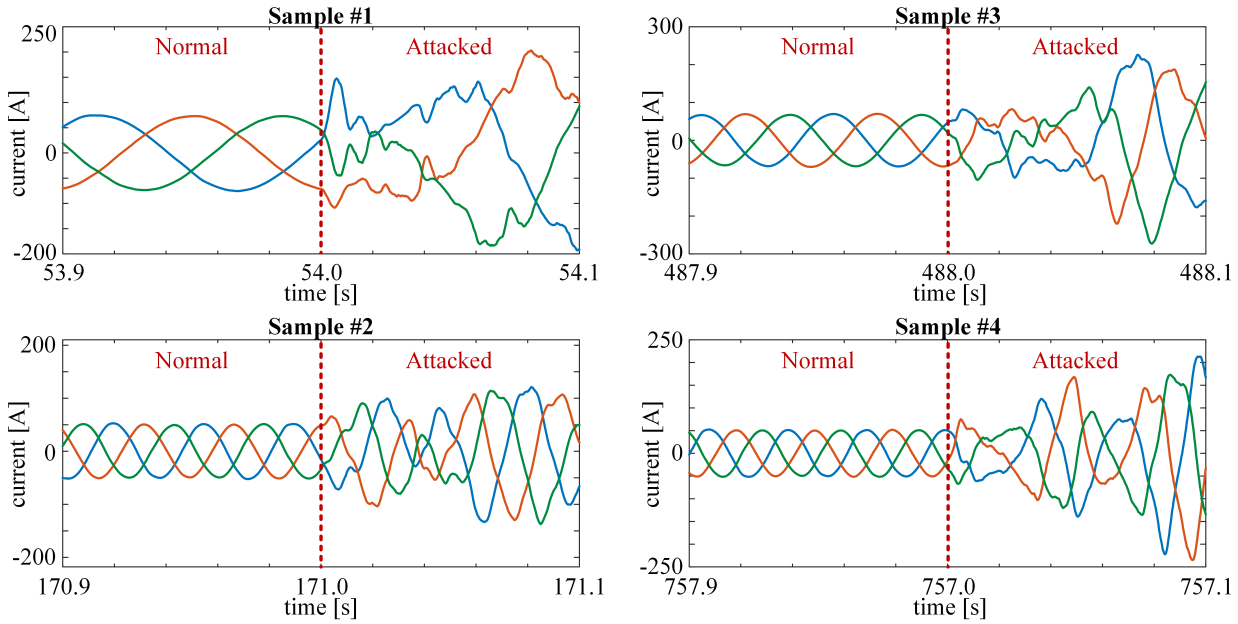


Figure 4.11: Samples of the motor three-phase current waveform from the case studies.

4.1.3 Method Demonstration with Real-Time Simulation

Considering the potential risk of conducting cyber threat experiments on real world EV powertrain testbed, we use the OPAL-RT real-time simulation instead to demonstrate and validate the proposed detection method. Fig. 4.11 shows 4 samples of the motor three-phase current waveform during the attacks in our case studies. In order to fully emulate the real world EV powertrain operation conditions,

a detailed real-time EV powertrain model is adopted. Fig. 4.10 shows the powertrain structure used in the HIL real-time simulation. The model includes two IPM traction motor drives at the front axis and the rear axis, respectively. In addition, vehicle mechanical systems including reduction gear box, shaft stiffness, tire-road interactions; and vehicle dynamics including road conditions, aerodynamics are also considered in the real-time simulation models. The detail descriptions of this testbed is elaborated in (Yang, Guo, & Ye, 2020). Meanwhile, the vehicle model is tested in the NEDC driving cycles shown in Fig. 4.12 and 40 cyber threat cases described in Table 4.1. The case studies focus on the d-axis and q-axis current references of the 2 traction motors (totally 4 target signals). In order to emulate the behavior of a typical false data injection attack, we adopt a random walk time series model, which is shown in Eq.(4.7), where the true data and false data are denoted as y and $\hat{y}$, respectively, and the attack period is $\mathbf{T_{ATK}} = [t_0, t_0 + T_a]$. R is a normal distribution random variable with zero mean and 0.5 variance. ($R \sim \mathcal{N}(0, 0.5)$)

$$\hat{y_k} = \begin{cases} y_k & (t \notin \mathbf{T_{ATK}}) \\ y_{k-1} + R & (t \in \mathbf{T_{ATK}}) \end{cases} \tag{4.7}$$

From the real-time simulation of the 82 case studies among three standard vehicle testing driving cycles, New European Driving Cycle (NEDC), Urban Dynamometer Driving Schedule (UDDS), and Highway Fuel Economy Test Cycle (HWFET). We then generate 200 windows of abnormal observations from each case, thus totally 16400 observations, labeled as "attack". Among the 16400 abnormal observations, we randomly choose 13120 (80%) of them to participate the training process so that during the testing, there will be 3280 new observations which doesn't participate in the training. Then, we also randomly select 13120 observations from the normal conditions to participate the training process. In order to demonstrate the performance of the classifier candidates, we conduct an extra testing with all the 82 attack cases to calculate their corresponding time to detect. In addition we also use the 3208 normal observations from the testing data sets to test the model false alarm rates.

In the following sections, the proposed method is demonstrated and validated from three aspects: performance of the proposed method in vehicle driving cycles; evaluation of online memory savings with window size comparison; and time-to-detect of different binary classifiers.

Performance of the proposed method based on testing data sets

For demonstration purpose, we choose four binary classifiers as described in section III. In Table. 4.2, the accuracy, the κ statistics, and the 95% confident interval of four binary classifier candidates based on the testing data are listed, the confusion matrices are shown in Fig. 4.13, and the histogram of these results is depicted in Fig. 4.14 for a clear comparison. According to these results, the classification accuracy of all four classifiers are above 95%, which suggests that the time-domain features extracted from the d-axis and q-axis currents are able to distinguish most of the attack conditions from the normal conditions. Furthermore, Random Forests and AdaBoost Classification Tree classifiers achieve an accuracy above 99%, which suggests that these two classifiers are highly sensitive to the anomalies reflected by the extracted features and could detect almost all the abnormal observations.

Table 4.1: Lists of Case Studies

Case No.	Driving Cycle	Attack Time (s) (attacks last 0.5 seconds)
1-40	NEDC	[54 98 171 250 289 366 488 573 758 873]
41-80	UDDS	[23 168 357 458 523 572 647 777 1058 1169]
81-82	HWFET	[320]

Table 4.2: Performance Statistics of the Binary Classifier

	RF	SVM	KNN	AdaBoost
Accuracy	0.9979	0.9809	0.9562	0.9994
κ	0.9957	0.9619	0.9125	0.9988
95% Confident Interval (lower)	0.9964	0.9773	0.9510	0.9984
95% Confident Interval (upper)	0.9988	0.9841	0.9611	0.9998

Table 4.3: False Alarm Test Results Among 3280 Normal Observations

Detect Results	RF	SVM	KNN	AdaBoost
Normal	3266	3163	3163	3278
Attack	14	117	117	2
False Alarm Rate (%)	0.43	3.57	3.57	0.06
Normal (with Majority Votes)	3280	3280	3280	3280
Attack (with Majority Votes)	0	0	0	0

Meanwhile, Table 4.3 shows the results of the false alarm rates among the 3280 normal observations that do not participate the training process. From Table 4.3, random forest classifier and the AdaBoost classifier could achieve a false alarm rate less than 0.5% while the other two can achieve a false alarm rate less than 4%. However, this results only show the performance of the classifiers, if including the assistance of the majority vote mechanism, the false alarm of all four binary classifiers will be completely eliminated. However, the majority vote mechanism will increase the time to detect the anomalies and the computational resources as well. In our simulations, we adopt a voting mechanism with vote capacity of 6.

In addition, impacts of the distorted position signals are discussed as follow. First of all, as discussed in section II, the reason we assume d- and q- axis current signals are secure is that these two signals could be calculated from the three-phase measurements and the rotor position directly. More specifically, the three-phase measurements could be acquired from the original sensors or extra third-party sensors. Both ways could easily guarantee the security of the current signals. On the other hand, the rotor position could be acquired from the original rotor encoders or estimated from the three-phase currents. Therefore, if we could guarantee the security of three-phase current signals, we could assume the position is secure.

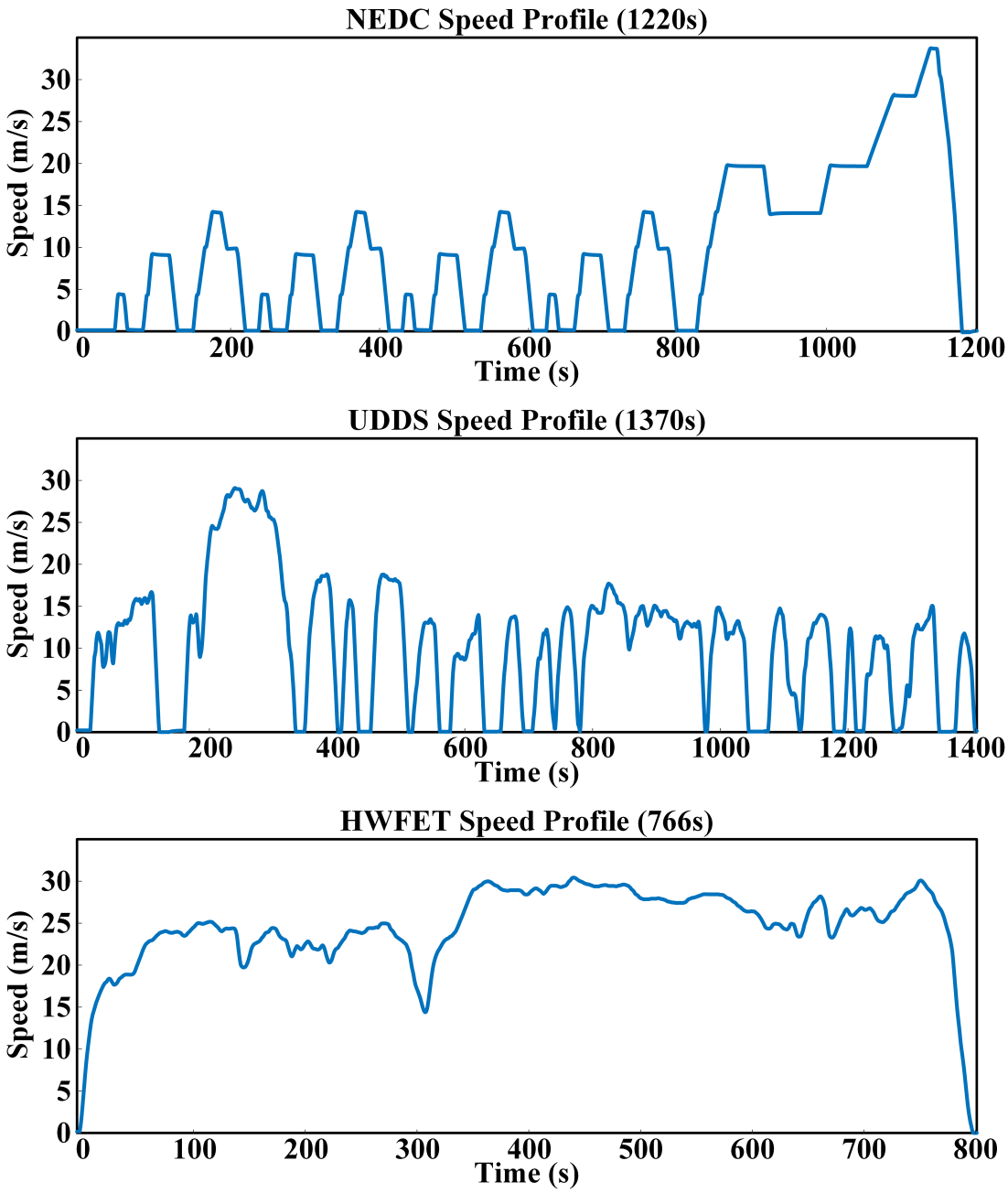


Figure 4.12: Plots of the driving cycles.

Secondly, if the position signal gets distorted, the motor drive controller will receive incorrect d- and q-axis current feedback. Then, the actual currents fed to the detector will be deviated from the normal

region and the proposed detector will be able to detect these kinds of anomalies. Therefore, the proposed detection method will still work even if the position signal gets distorted.

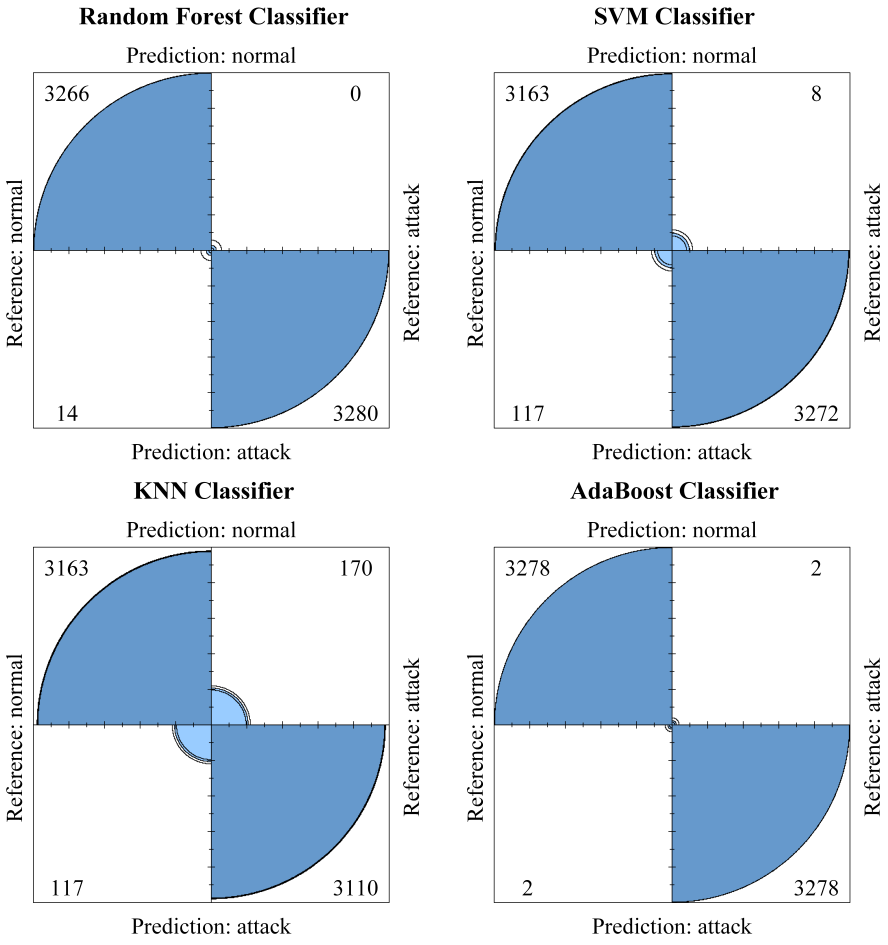


Figure 4.13: Confusion matrix of testing results of the four binary classifiers.

4.1.4 Evaluation of computational resource savings with window size comparison

In order to evaluate the computational resource savings, we compare the proposed methods with traditional spectrum-based methods from two aspects: online memory savings and computational complexity. First, to compare the online memory savings, we use the monitoring window size as the evaluation criteria, because saving observations occupies most of the online memory and the larger window size required by the algorithms, the more online memory is required. As stated in (Thomson & Fenger, 2001), using CSA require a high resolution frequency spectrum. Commonly, the frequency resolution for CSA is less than 10Hz, therefore, Table 4.4 lists the window size requirement for the proposed method and the CSA with different frequency resolutions. The results are based on a sampling time of 0.2ms (5 kHz sampling

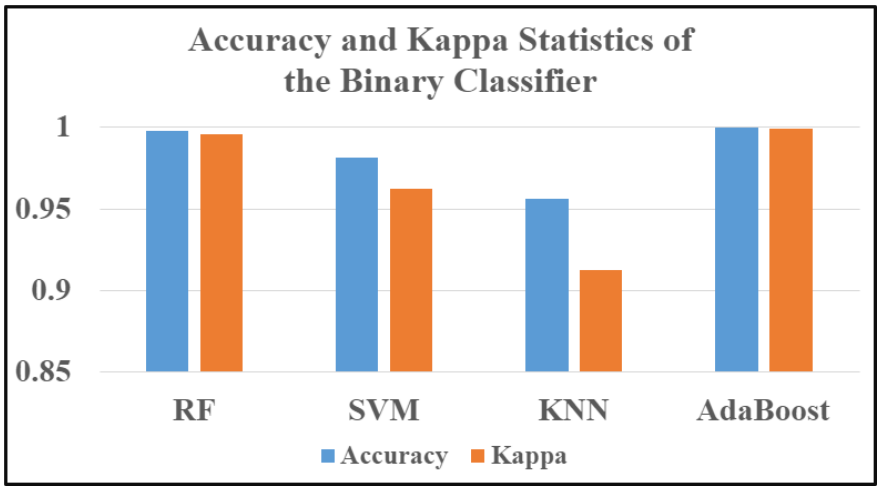


Figure 4.14: Accuracy and κ statistics of the binary classifiers.

frequency). As the results suggest, even the frequency resolution is 10Hz, the window size requirement is still twice of the one for the proposed method. Therefore, the proposed method will save huge amount of online memory. In addition, from the aspect of computational complexity, the proposed method requires linear computational time, $O(n)$, because it is based on the calculation of mean, variance, skewness, kurtosis. Meanwhile, the spectrum-based method requires quadratic computational time, $O(n^2)$. Apparently, the proposed method require less online computational resources than the traditional spectrum-based methods.

Table 4.4: Window size requirement for the proposed algorithm and the CSA with different frequency resolutions

	Proposed Method	Traditional CSA		
Resolution	NA	2Hz	5Hz	10Hz
Size (points)	500	5000	2000	1000
Size (s)	0.1	1.00	0.40	0.20

Time-to-detect of different binary classifiers

As the time-to-detect of anomalies depends on many external factors, such as hardware computational performance, sensor sampling frequency, we compare the time-to-detect of the proposed method with traditional spectrum-based methods by comparing the number of sliding windows required to generate an anomaly alarm. For the proposed methods, the detection accuracy of random forest classifiers and AdaBoost classifier could achieve as high as 99.9%. This means that detector based on these two classifiers could immediately generate an anomaly alarm, even without majority vote mechanisms. Therefore, for

these detector, the number of sliding windows required is 1. The other candidate binary classifiers have lower accuracy; therefore, they need the assists of majority vote mechanisms. Among them, the KNN-classifier-based detector has the worst performance, and it requires vote capacity of 6 to achieve 100% accuracy. This means that the proposed method at most requires 6 sliding windows to generate an accurate anomaly alarm. On the other hand, considering the high sampling rates of the traditional spectrum-based methods, they require much longer observations to reflect the attack impact on the spectrum. Therefore, in our case studies, the spectrum-based methods requires at least 20 sliding windows to generate an accurate anomaly alarm. These results support that the proposed method could detect those anomalies in a lot shorter time.

4.2 Detection and Diagnosis of Physical Faults and Stealthy Cyberattacks in Dual-Motor EV Powertrains with Data-Driven Motor Current Signature Analysis

With the pervasive utilization of digital control units and communication networks in the modern electric vehicle powertrains, such safety-critical systems become highly vulnerable to potential cyber threats. In 2010, Koscher et al. experimentally evaluated the cyber-physical security issues on a modern automobile and demonstrated the fragility of the underlying system structure (Koscher et al., 2010).

As the traction motor drive is one of the most critical components in an EV powertrain, its safety and reliability are always priorities during design, implementation, and maintenance. So far, there has been much literature on the safety and security of motor drive systems. Among this literature, the proposed anomaly detection and root-cause diagnostic methods lie in model-based and data-driven methods. Most model-based methods use time-domain or frequency-domain signals. For example, fast Fourier Transform (FFT) is used in monitoring steady-state conditions in the frequency domain, while short-time Fourier Transform (STFT) is used in fluctuating load and speed conditions. Other examples are spectrograms and time-frequency analysis using wavelets and Wigner-Ville transforms. Usually, the system current, flux, mechanical vibration, torque, and speed signals are analyzed. One of the most widely used frequency-domain methods is motor current signature analysis (MCSA), based on the FFT of system line currents. Such methods usually pre-locate characteristic frequencies for specific fault scenarios and then use these frequency components' magnitudes and phase angles for detection and diagnosis. (Kar & Mohanty, 2006; Kliman & Stein, 1992; Thomson & Fenger, 2001) Meanwhile, time-domain methods usually rely on pre-defined residuals. These methods compare the system outputs and the sets of reference values. The reference values could be acquired from two primary approaches. One comes from the signal itself, such as the average or limit values. The other one often relies on pre-defined system models from prior system knowledge. These models calculate the reference values by predicting the system outputs based on the given inputs at each instant. Then, the differences between actual outputs and reference values are defined as residuals. Alarms will be generated when such residuals exceed some pre-defined thresholds. (Dexter, 1995; Eldin et al., 1994; Kim & Parlos, 2002) On the other hand, most data-driven approaches

follow a similar framework, which includes three primary tasks: (1) a preprocessing task where input data from sensors and logs is normalized and organized for further analysis, (2) a pre-defined anomaly detector that analyzes the system status, (3) a pre-trained classifier that, based on the current system status and monitoring signals, provides a diagnosis for the system. (Bo et al., 2019) By following this framework, (Anwar et al., 2015) utilized a statistical learning approach to differentiate the physical faults from cyber-attacks based on data generated by the IEEE 30 bus benchmark test system; Furthermore, (Khan et al., 2021) proposed an intelligent anomaly identification (IAI) technique for inverter-based systems by utilizing data-driven artificial intelligence tools that employ multi-class support vector machines (MSVM) for anomaly classification and localization. Meanwhile, Ye et al. researched the threat detection and attack resilient control from both global vehicle control level and local motor drive control level in (Guo, Yang, & Ye, 2021; Guo, Yang, Ye, & Velni, 2021; Guo, Yang, Ye, Velni, & Song, 2021; Guo, Ye, & Yang, 2021; Yang, Guo, Li, et al., 2020a; Yang, Guo, & Ye, 2021a; Yang, Ye, & Guo, 2021; J. Ye et al., 2021).

Nevertheless, most current literature only addresses cyber-security or physical safety individually instead of simultaneously. (D. Zhang et al., 2021) In addition, most studies have focused on really aggressive attacks. Such attacks usually will cause drastic changes and disturbances to the systems, which makes these attacks easier to detect. Meanwhile, little of the current research has addressed the importance of distinguishing between cyberattacks and physical faults.

This section proposed a data-driven approach to detect and diagnose cyberattacks and physical faults in the dual-motor electric vehicle powertrain. The proposed method uses the motor line current spectra and four widely used data-driven classification methods to detect anomalies and distinguish the stealthy cyber-attacks from common physical faults. In addition, we used motor-bearing faults, inter-turn short circuit faults, and false data injection attacks as case studies to verify the performance of the proposed method.

4.2.1 Models for Physical Faults and Cyberattacks

To validate the proposed approach, we first formed a comprehensive case study in a dual-motor powertrain system including one permanent magnet synchronous machine (PMSM) and one induction machine (IM). Fig. 4.10 shows a general diagram of the system. The case study includes motor-bearing faults in IM, inter-turn shorts faults in PMSM, and false data injection attacks on controllers of both machine drives. In this section, the dynamic models of these faults and attacks are discussed in detail.

Physical Faults: Bearing Faults in IM

Bearing faults are one of the most common physical failures in electric machines. According to (Toliyat et al., 2012), bearing faults account for more than 40% of all the electric motor failures. Therefore, it is one of the best candidates to study the differences between physical faults and cyberattacks. When a bearing fault appears in the machine, some periodic vibration pulses will be generated due to the impact among the rolling elements, the bearing raceways, and the cage. The periodic pulses have different characteristic frequencies depending on the fault types. As demonstrated in (S. Zhang et al., 2020), there are primarily

five types of common bearing faults, and these different bearing faults will generate some periodic vibration pulses due to the impact among the rolling elements. The types and characteristic frequencies of these bearing faults are shown below, where the number of balls is denoted as N_b, the ball diameter is d, and the pitch or cage diameter is D. The point of contact between the ball and the raceway is characterized by the contact angle θ, and f_r is the mechanical frequency of the rotor. The geometry parameters of the bearing are shown in Fig. 4.15.

1. Cage defect hits the outer raceway:
   $f_{co} = \frac{f_r}{2}(1 - \frac{d}{D}\cos\theta)$;
2. Cage defect hits the inner raceway:
   $f_{ci} = \frac{f_r}{2}(1 + \frac{d}{D}\cos\theta)$;
3. Outer raceway defect hits balls:
   $f_o = N_b\frac{f_r}{2}(1 - \frac{d}{D}\cos\theta)$;
4. Inner raceway defect hits balls:
   $f_i = N_b\frac{f_r}{2}(1 + \frac{d}{D}\cos\theta)$;
5. Ball defect hits both raceways:
   $f_b = \frac{D}{d}f_r(1 - \frac{d^2}{D^2}\cos^2\theta)$.

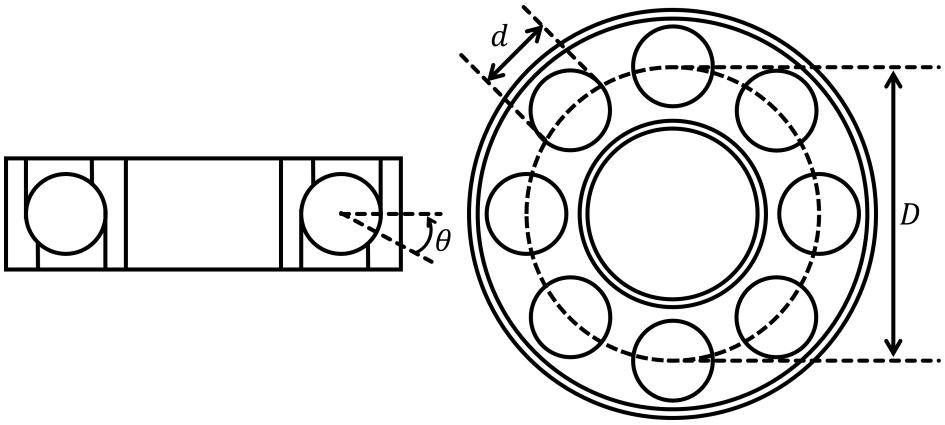


Figure 4.15: Geometry parameters of the bearing.

The periodic pulses caused by vibration will then introduce geometry asymmetry to the machine. Such asymmetry will then change the inductance of the machine. Ideally, the inductance variation should be composed of an infinite number of characteristic frequency harmonics. To simplify the model, we only choose the fundamental frequency component to represent the inductance variation. Then, we introduce a fault inductance to the induction machine state-space model in Eq. (4.8) and Eq. (4.9), where u is the stator input voltage, R and L are the motor resistance and inductance, ω_r is the electrical rotor speed, λ is the flux linkage, subscripts s, r, and m denote the stator, rotor, and their mutual electromagnetic

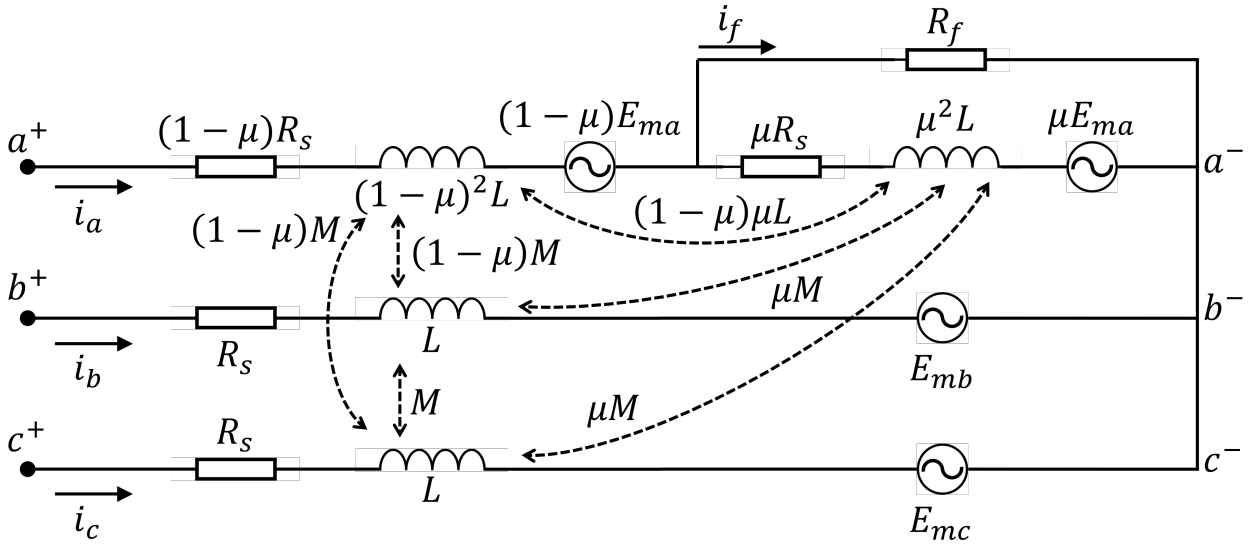


Figure 4.16: The equivalent circuit of the ITSC fault.

parameters. Eq. (4.10) shows the new inductance after the bearing fault appears, where L_m^{fault} denotes the fault inductance, ω_c is the characteristic frequency described above, and ΔL_m is the inductance variation magnitude describing the fault severity.

$$\begin{cases} u_{ds} &= R_s i_{ds} + \frac{d}{dt}\lambda_{ds} - \omega_s \lambda_{qr} \\ u_{qs} &= R_s i_{qs} + \frac{d}{dt}\lambda_{qs} + \omega_s \lambda_{dr} \\ 0 &= R_r i_{dr} + \frac{d}{dt}\lambda_{dr} - (\omega_s - \omega_r)\lambda_{qr} \\ 0 &= R_r i_{qr} + \frac{d}{dt}\lambda_{qr} + (\omega_s - \omega_r)\lambda_{dr} \end{cases} \tag{4.8}$$

$$\begin{cases} \lambda_{ds} &= L_s i_{ds} + L_m i_{dr} \\ \lambda_{qs} &= L_s i_{qs} + L_m i_{qr} \\ \lambda_{dr} &= L_m i_{ds} + L_r i_{dr} \\ \lambda_{qr} &= L_m i_{qs} + L_r i_{qr} \end{cases} \tag{4.9}$$

$$L_m^{fault} = L_m + \Delta L_m \cos \omega_c t \tag{4.10}$$

Physical Faults: Inter-Turn Short Circuit Faults in PMSM

The inter-turn short circuit (ITSC) fault is caused by partially short circuit faults in some turns of stator windings due to isolation aging or failures. Such aging and failures are usually caused by line current harmonics and chemical corrosion. When the ITSC faults appear, a local loop will be generated in the fault location. This loop will induce a high-amplitude short circuit fault current and cause local overheat.

Then, further damage to the machine windings. In this section, the ITSC fault is modeled in the context of permanent magnet synchronous machines. The equivalent circuit of such ITSC fault is shown in Fig. 4.16. (Suppose the fault appears in phase A.) In Fig. 4.16, there are n turns out of N-turn-winding shorted. The fault turn ratio μ is then defined as $\mu = \frac{n}{N}$. In addition, the short circuit contact resistance is defined as R_f, and i_f is the circulating current caused by the short circuit fault.

According to the equivalent circuit in Fig. 4.16, the system equations under ITSC fault could be written as followings:

$$v_{a,f} = R_s \cdot i_{a,f} - \mu R_s \cdot i_f + \frac{d}{dt}\lambda_{a,f} \tag{4.11}$$

$$v_{b,f} = R_s \cdot i_{b,f} + \frac{d}{dt}\lambda_{a,f} \tag{4.12}$$

$$v_{c,f} = R_s \cdot i_{b,f} + \frac{d}{dt}\lambda_{a,f} \tag{4.13}$$

$$\begin{aligned}\lambda_{a,f} =& L_{aa} \cdot i_{a,f} + M_{ab} \cdot i_{b,f} + M_{ac} \cdot i_{c,f} \\ & - \mu L_{aa} \cdot i_f + \lambda_{PM} \cdot \cos(\theta)\end{aligned} \tag{4.14}$$

$$\begin{aligned}\lambda_{b,f} =& M_{ba} \cdot i_{a,f} + L_{bb} \cdot i_{b,f} + M_{bc} \cdot i_{c,f} \\ & - \mu M_{ba} \cdot i_f + \lambda_{PM} \cdot \cos(\theta - \frac{2\pi}{3})\end{aligned} \tag{4.15}$$

$$\begin{aligned}\lambda_{c,f} =& M_{ca} \cdot i_{a,f} + M_{cb} \cdot i_{b,f} + L_{cc} \cdot i_{c,f} \\ & - \mu M_{ca} \cdot i_f + \lambda_{PM} \cdot \cos(\theta + \frac{2\pi}{3})\end{aligned} \tag{4.16}$$

$$v_f = 0 = -(R_f + \mu R_s) \cdot i_f + \mu R_s \cdot i_{a,f} + \frac{d}{dt}\lambda_f \tag{4.17}$$

$$\begin{aligned}\lambda_f =& \mu L_{aa} \cdot i_{a,f} + \mu M_{ab} \cdot i_{b,f} + \mu M_{ac} \cdot i_{c,f} \\ & - \mu^2 L_{aa} \cdot i_f + \mu \lambda_{PM} \cdot \cos(\theta)\end{aligned} \tag{4.18}$$

Cyberattacks: False Data Injection Attacks

The cyberattack in this case study is also one of the most common attacks, the false data injection (FDI) attack. It means the attacker maliciously injects false data sequences into the digital control units after compromising, such as wrong parameters, abnormal register status, Etc. In this section, the case study of FDI attacks includes two types of false data injections: (1) bias injections to the motor line current feedback signals; (2) malicious control sequences injected into the motor voltage commands. These attacks are designed to mimic the behaviors of ITSC and bearing faults discussed in previous sections. The goal of such attacks is to confuse the original fault detectors and cause unnecessary downtime and maintenance costs. The attack model is shown in Eq. (4.19) and Eq. (4.20), where $\hat{s_k}$ is the attacked signal, s_k is the original signal, K_{atk} is the bias coefficient, $M \cdot \sin \omega_{atk} t$ is the malicious control sequences, and ω_{atk} is selected according to the characteristic frequencies of bearing faults. The attack period is denoted as

$\mathbf{T_{ATK}}$.

$$\text{Type I: } \hat{s_k} = \begin{cases} s_k & (t \notin \mathbf{T_{ATK}}) \\ s_k \cdot K_{atk} & (t \in \mathbf{T_{ATK}}) \end{cases} \tag{4.19}$$

$$\text{Type II: } \hat{s_k} = \begin{cases} s_k & (t \notin \mathbf{T_{ATK}}) \\ s_k + M \cdot \sin \omega_{atk} t & (t \in \mathbf{T_{ATK}}) \end{cases} \tag{4.20}$$

4.2.2 Data-Driven Approach for Detecting and Diagnosing Physical Faults and Cyberattacks

The proposed approach is to search for features and patterns in the motor line current spectra, which could distinguish the system status among the healthy, fault, and attack conditions. As shown in Fig. 4.17 - Fig. 4.19, the line current waveforms of these three statuses are highly similar and are not distinguishable via human eyes. Therefore, the traditional rule-based methods through establishing some pre-defined thresholds are not feasible in these situations. The proposed approach adopts and compares four data-driven classification methods, namely random forests (RF), logistic regression (LR), support vector machines (SVM), and k-nearest-neighbor (KNN). These four classification methods represent four of the most widely used approaches for classification:

1. **Random Forests** (RF) is a modified decision tree-based classifier, which operates by constructing a multitude of decision trees at training and outputting the dominant class among all the classes generated from each decision tree.

2. **K-Nearest-Neighbor** (KNN) classifies the new observation by a plurality vote of its neighbors in the feature space and assigns the new observation to the most common class among its k nearest neighbors.

3. **Support Vector Machines** (SVM) is a representation of the training data set as support vectors (or points) in the feature space, which map the training data set to the separate classes divided by a clear gap; and new observations are then mapped into the same space and predicted to belong to the class based on the side of the gap on which they fall.

4. **Logistic Regression** (LR) models the probability function of a certain class based on the predictors. For example, $l = \log_b \frac{p}{1-p} = \beta_0 + \sum_{i=1}^{q} \beta_i x_i$

The proposed method will monitor the motor line current signals using sliding windows. The window size will be determined by the signal sampling frequency and required spectrum frequency resolution. The proposed method will first extract the signal spectra using fast Fourier transformation at every instant. Then, it will feed the frequency features to the pre-trained classification models. According to the current and previous classification results, a majority vote mechanism will be applied to determine the final detection outcomes. The detection results will then determine whether or not to generate an alarm. The detailed algorithm is shown in Algorithm 3.

Algorithm 3 Anomaly Detection Algorithm for Distinguishing Stealthy Cyber-Attacks from Common Physical Faults

Input: Real time line current measurements of both motor drives.
Output: Detection and diagnostic alarms.
Setting total voting capacity as v_{max};
for $i = 0, 1, 2, 3, 4, ...$ **do**
  for $k = 0, 1, 2, 3, ..., v_{max}$ **do**
    Recording and storing real-time measurements in a sliding window with size m and sampling time t_s;
    Extracting frequency features using FFT;
    Calculating the classification results by feeding the frequency features into pre-trained data-drive classification model;
    Saving the classification result D_k;
  end for
  Selecting the dominant classification result in D as A_i;
  Ouptut detection and diagnostic alarms A_i.
  Clearing the voting array D;
end for

4.2.3 Case Study and Simulation Results

In this section, the case study adopted a dual-motor electric vehicle powertrain to simulate the fault and attack scenarios to validate the performance of the proposed method. The general diagram of the powertrain is shown in Fig. 4.10. The powertrain consists of two motor drives: the front-wheel-drive is a permanent magnet synchronous machine (PMSM), and the rear-wheel-drive is an induction machine (IM). Both machines are controlled by Filed-Oriented-Control (FOC) with proportional-integral (PI) regulators. The detector extracts the motor line winding current signals from both front and rear drives. The sampling rate is 20 kHz, and a sliding window of size 400 is adopted. Then, the detector calculates the spectra from both current signals and extracts the magnitudes for frequencies ranging from 0 Hz to 1000Hz.

Table 4.5: List of Steady-State Operating Conditions

No.	Machine Speed (rpm)	Load Torque (Nm)
1	1000	200
2	1500	150
3	1000	50
4	1500	50

The case study includes 128 scenarios covering different bearing faults, ITSC faults, FDI attacks, and operating conditions. The details of these scenarios are described in TABLE 4.5 - TABLE 4.8, and some samples of line current waveforms in the case study are shown in Fig. 4.17 - Fig. 4.19. According to the waveforms in Fig. 4.17 - Fig. 4.19, the cyber-attacks and physical faults are hard to distinguish using traditional residual-based methods.

Table 4.6: List of ITSC Fault Scenarios

No.	Short-Turn Ratio μ	Short Resistance R_f
1	0.10	1
2	0.15	1
3	0.20	1
4	0.25	1
5	0.10	5
6	0.15	5
7	0.20	5
8	0.25	5

Table 4.7: List of Bearing Fault Scenarios

No.	Characteristic Frequency	$\Delta L_m/L_m$
1	f_{co}	0.4
2	f_{ci}	0.4
3	f_o	0.4
4	f_i	0.4
5	f_{co}	0.6
6	f_{ci}	0.6
7	f_o	0.6
8	f_i	0.6

Table 4.8: List of FDI Attack Scenarios

No.	Bias Injection K_{atk}	Malicious Control
1	+0.1	NA
2	-0.1	NA
3	+0.2	NA
4	-0.2	NA
5	NA	$M = 8, f_{atk} = f_{co}$
6	NA	$M = 8, f_{atk} = f_{ci}$
7	NA	$M = 8, f_{atk} = f_o$
8	NA	$M = 8, f_{atk} = f_i$

With the 128 scenarios in the case study, we generated 12800 samples of the line current frequency features. Among these samples, 80% are randomly selected as the training data sets, and the rest 20% are the testing sets.

The testing results and detection accuracy of the four classification methods are shown in TABLE 4.9 - TABLE 4.13. In addition, normalized confusion matrices of each method are shown in Fig. 4.20 as well.

Among these results, all the classification methods could achieve an accuracy above 80% apart from the Logistic Regression, and Random Forests could reach an accuracy higher than 90%. Meanwhile, according to the confusion matrices and detection outcomes, distinguishing such cyber-attacks from common physical faults is much more challenging than distinguishing abnormal conditions from normal conditions. For all four methods, the accuracy of detecting anomalies is almost 100%.

Table 4.9: Accuracy of Data-Driven classifiers

Random Forests	90%
K-Nearest-Neighbor	84%
Support Vector Machine	82%
Logistic Regression	63%

Table 4.10: Confusion Matrix of Testing Reuslts: KNN

		Prediction		
		Attack	Normal	Fault
Reference	Attack	345	37	273
	Normal	0	1264	0
	Fault	64	42	535

Table 4.11: Confusion Matrix of Testing Reuslts: LR

		Prediction		
		Attack	Normal	Fault
Reference	Attack	211	404	40
	Normal	7	1257	0
	Fault	51	451	139

Table 4.12: Confusion Matrix of Testing Reuslts: RF

		Prediction		
		Attack	Normal	Fault
Reference	Attack	494	0	161
	Normal	0	1264	0
	Fault	69	4	568

Table 4.13: Confusion Matrix of Testing Reuslts: SVM

		Prediction		
		Attack	Normal	Fault
Reference	Attack	374	150	131
	Normal	0	1264	0
	Fault	79	112	450

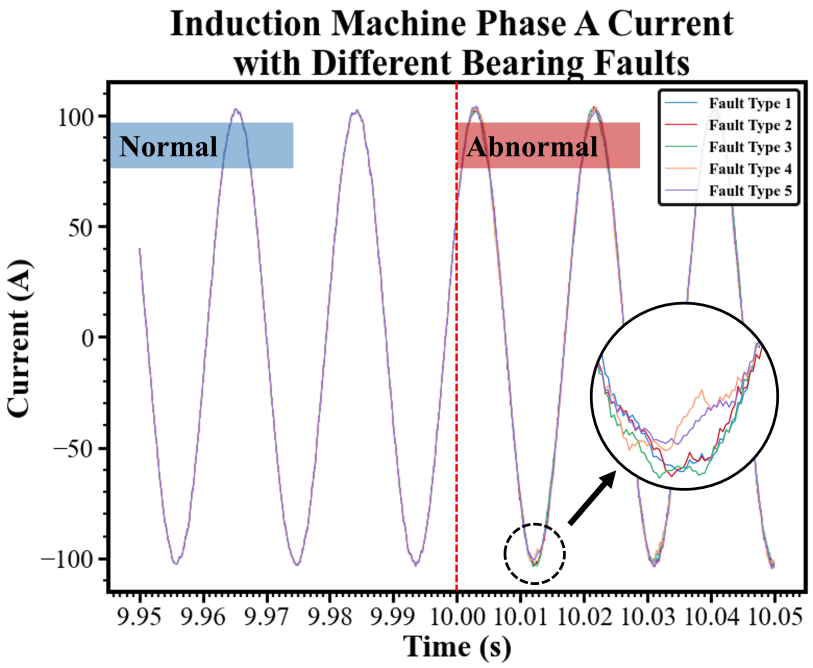


Figure 4.17: Samples of the line current waveforms with different bearing faults in the induction machine.

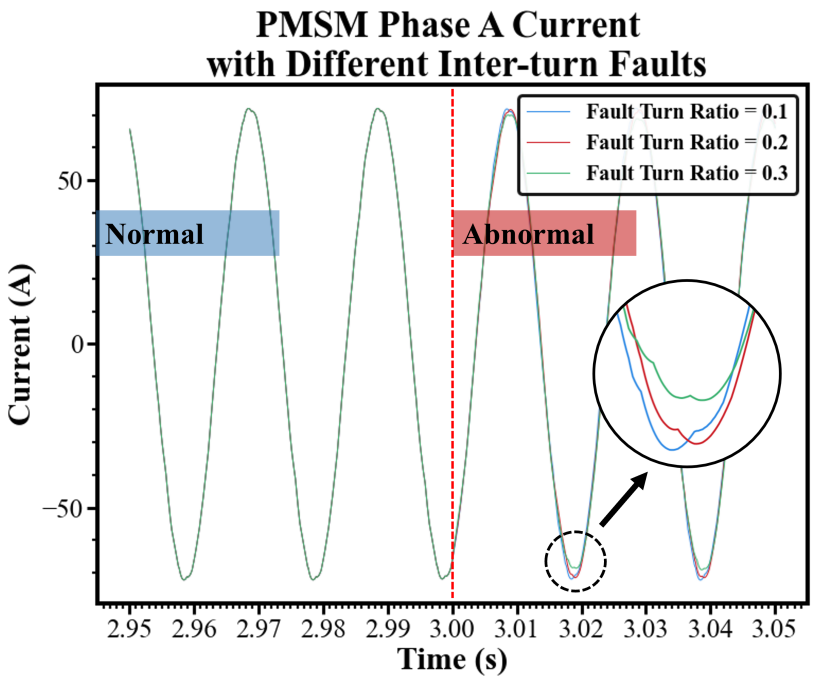


Figure 4.18: Samples of the line current waveforms with different ITSC faults in the PMSM.

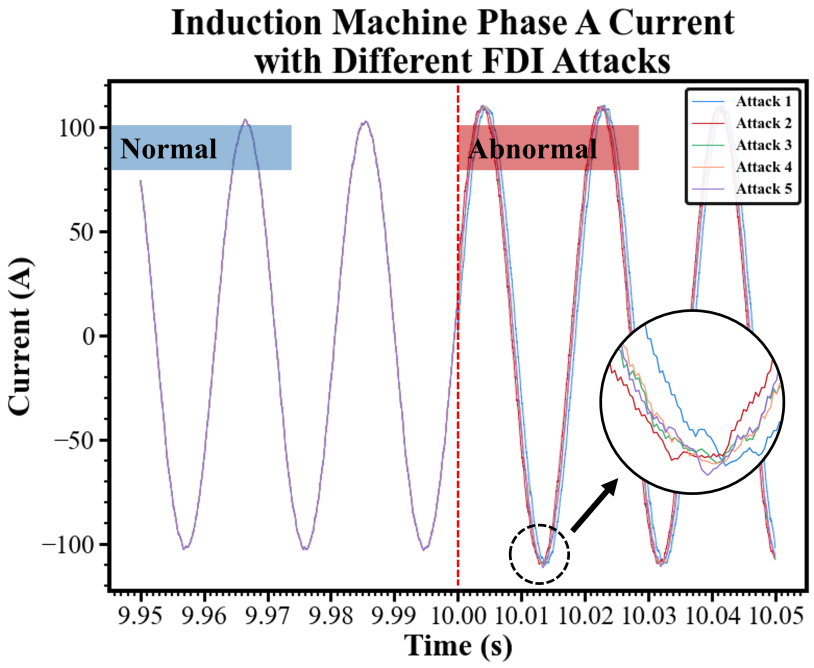


Figure 4.19: Samples of the line current waveforms with different FDI attacks in the induction machine.

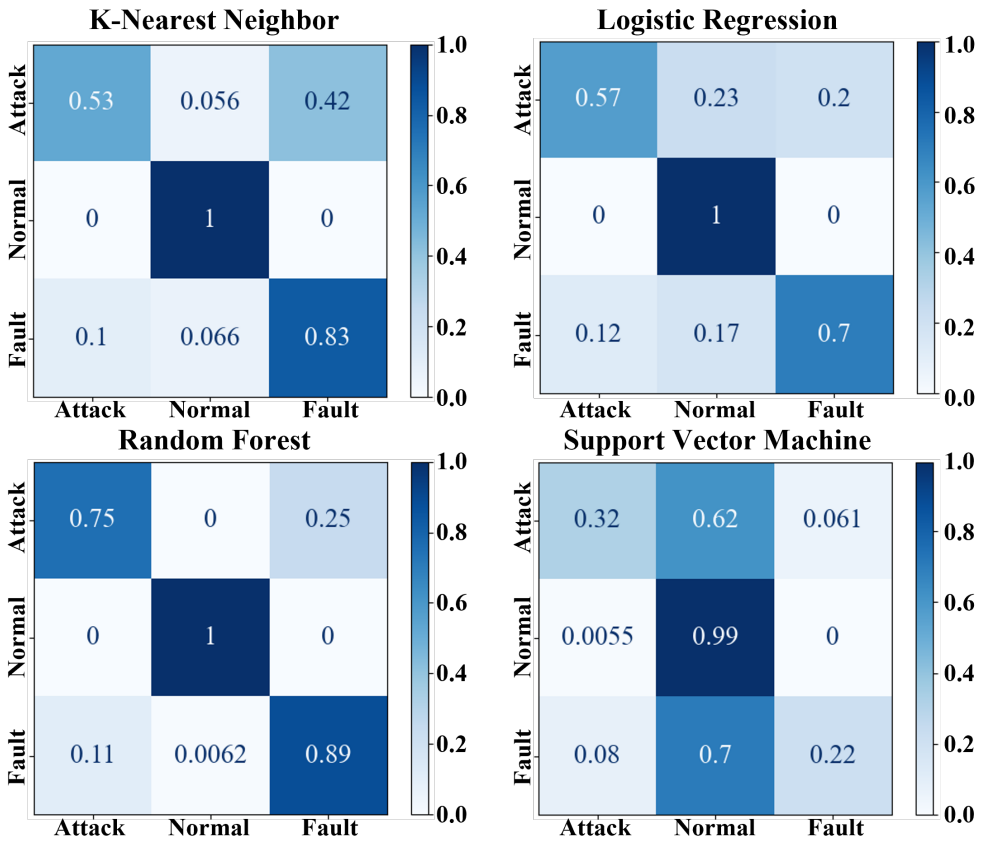


Figure 4.20: Confusion matrices of 4 data-driven classifiers.

4.3 Detection and Diagnosis of Physical Faults and Cyberattacks in Manufacturing Motor Drives with Limited Monitoring Sensors

Valuable and safety-critical systems are facing new threats from physical and cyber domains due to the pervasive utilization of digital control units and communication networks in modern manufacturing systems. There is an urgent need for future manufacturing systems to have an advanced monitoring solution targeting physical faults and cyber-attacks, especially for high-power motor drives. Gap still exists between studies of physical faults and cyber-attacks on motor drives. Such a gap could be summarized in three aspects. First, cyber-attacks and physical faults have different dynamics and mechanisms. Physical faults will essentially change part of the fundamental physics of the systems. For example, inter-turn short circuit faults will create a local short circuit loop within the fault winding. Moreover, bearing faults will cause machine shaft vibrations, leading to periodic changes in the machine winding inductances. However, cyber-attacks will not necessarily change the system's fundamental physics. Instead, they will directly or

indirectly change the digital controller behaviors to achieve some objectives established by attackers. The second gap is that cyber-attacks and physical faults have different countermeasures. For physical faults, the most common protocol is to shut down and isolate the fault system and conduct thorough maintenance to eliminate the faults. On the other hand, when the system detects cyber-attacks, some standard approaches include rebooting the system or conducting a hot patch. Finally, in most recent literature, cyber-attacks and physical faults are still two different topics in different communities. Therefore, most research focuses on physical faults or cyber-attacks individually instead of simultaneously.(D. Zhang et al., 2021) For motor drives, studies of physical faults are more mature than cyber-attacks. In the last decades,

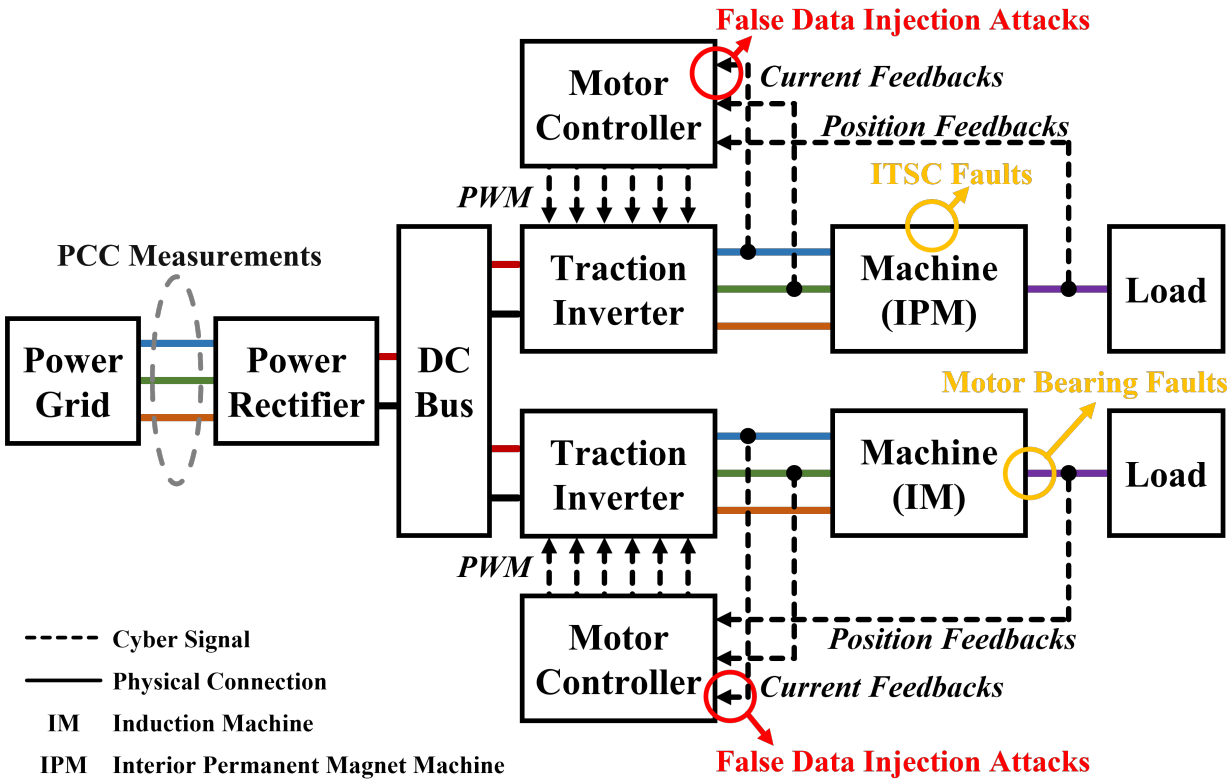


Figure 4.21: System diagram of the dual-motor network.

there have been many literatures addressing physical faults from different directions, such as motor current signature analysis(Kar & Mohanty, 2006; Kliman & Stein, 1992; Thomson & Fenger, 2001), time-domain analysis(Dexter, 1995; Eldin et al., 1994; Kim & Parlos, 2002), data-driven methods(Anwar et al., 2015; Bo et al., 2019; Khan et al., 2021), etc. Nevertheless, studies addressing cyber-attacks on motor drives did not arise until recent years. For example, (Yang, Guo, Li, et al., 2020a) studied the vulnerabilities of motor drives due to sensor attacks, and (Yang, Guo, & Ye, 2021b; Yang, Ye, & Guo, 2021) proposed a fast attack detection method based on motor time-domain and frequency-domain features, respectively. Meanwhile, most studies addressing cyber-attacks focused on general cyber-physical systems instead of motor drives. For example, (Cui et al., 2012; Kwon et al., 2013; Vuković & Dán, 2013) propoosed some

detection solutions using the attacker versus defender dynamics, distributed attack detection, and state recovery.

To guarantee system safety and security of thr future motor drives in manufacturing systems, it is critical to developing fast detection and accurate diagnostic solutions targeting cyber-attacks and physical faults. To narrow this gap, this section proposed a data-driven method for detecting and distinguishing cyber-attacks and common physical faults for manufacturing motor drives.

4.3.1 Data-Driven Detection and Diagnosis Method With Limited Sensors

The detection method proposed in this section is based on four different types of data-driven classifiers and uses only the line current signals from the Point of Common Coupling (PCC).

Fig. 4.22 shows a flowchart of the proposed method. The detection algorithm first acquires independent measurements from the isolated current sensors at the PCC. Then, the line current data is transformed to the frequency domain with Fast-Fourier-Transformation (FFT). After extracting and normalizing the spectrum features, the algorithm will then feed these features to four individual classifiers, namely k-nearest-neighbor (KNN), support vector machine (SVM), random forests (RF), and logistic regression (LR). These four classifiers will calculate the detection results independently. Finally, the algorithm will conduct a majority vote among the results generated from the classifiers, and generate an alarm when three or more classifiers detect a fault or an attack.

The rest of this section will discuss each step in more detail.

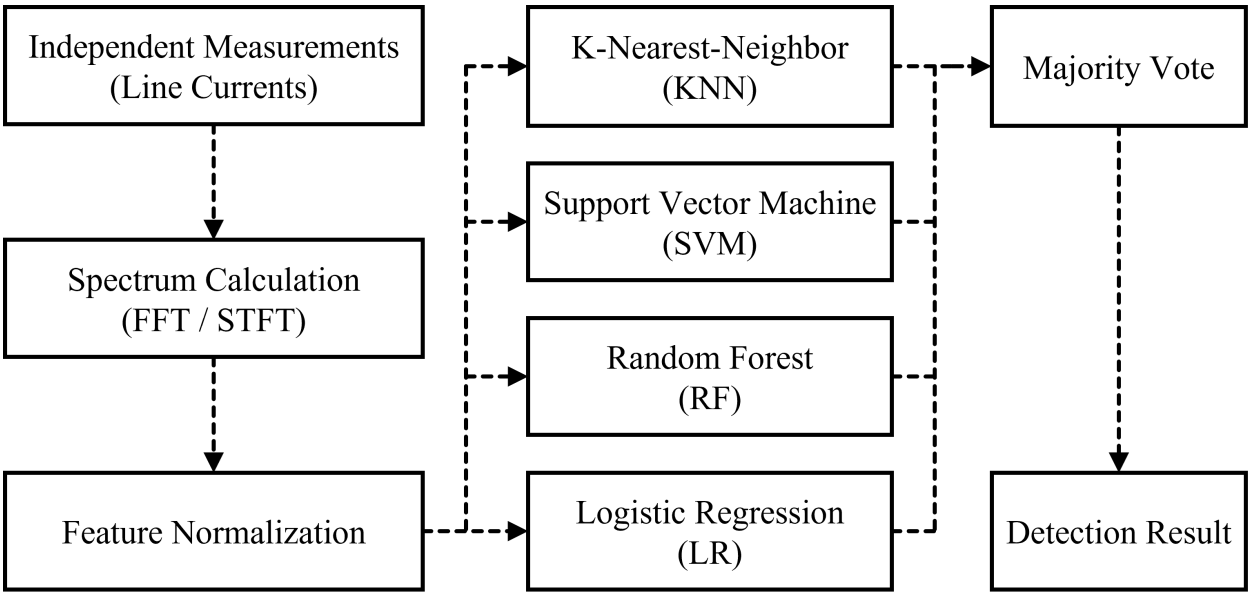


Figure 4.22: Flowchart of the proposed detection procedure.

Independent Measurements at PCC and Feature Extraction

The proposed method uses only the line current signals at the PCC for the following reasons:

1. It will be much easier to guarantee the safety and security of the monitoring system since it only requires one set of sensors at the PCC;
2. It will be possible to isolate the entire monitoring system from the original system and those vulnerable communication networks.
3. The monitoring system will be easy to install or upgrade due to its simplicity.
4. The overall cost of the monitoring system will be much lower than other solutions requiring extra information from individual machines.

In addition, it should be pointed out that the sampling frequency of these measurements should be no less than 2kHz because the maximum spectrum frequency required by the algorithm is at least 1kHz.

After acquiring the measured line current signals, the algorithm uses a sliding window to segregate the measurement data into batches. Within each sliding window, FFT is adopted to extract the current spectrum.

Data-Driven Classificatiers

Because the cyber-attacks targetting motor drive controllers could be highly sophisticated and it is challenging to distinguish such attacks from physical faults, using a single classifier will result in low detection accuracy. Therefore, this section adopted four different types of data-driven classifiers to improve detection accuracy. Each classifier processes the same features independently, and then a majority vote procedure will determine the final results based on the outcomes of each individual classifier. Such an approach will largely reduce the false detection results. The four classifiers represent four typical classification methods with different characteristics:

- RF represents classification tree based methods;
- KNN represents non-parametric classifiers;
- SVM represents support vector based classifiers;
- LR represents regression based algorithms.

4.3.2 Simulation Results

As stated in Section II, this section formed a comprehensive case study including physical faults and cyber-attacks in a dual-motor manufacturing motor drive network. Fig. 4.21 shows the system diagram of the simulation model for this dual-motor network. The system consists of two motor drives: a permanent magnet synchronous machine (PMSM) and an induction machine (IM). Both machines are controlled by Filed-Oriented-Control (FOC) with proportional-integral (PI) regulators on motor speed and current.

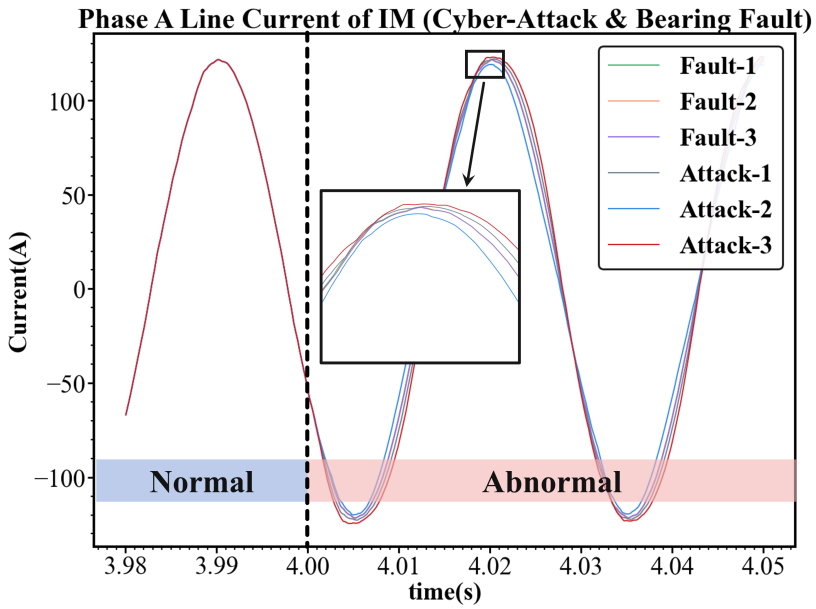


Figure 4.23: IM line current waveforms in different bearing faults and FDI attacks scenarios.

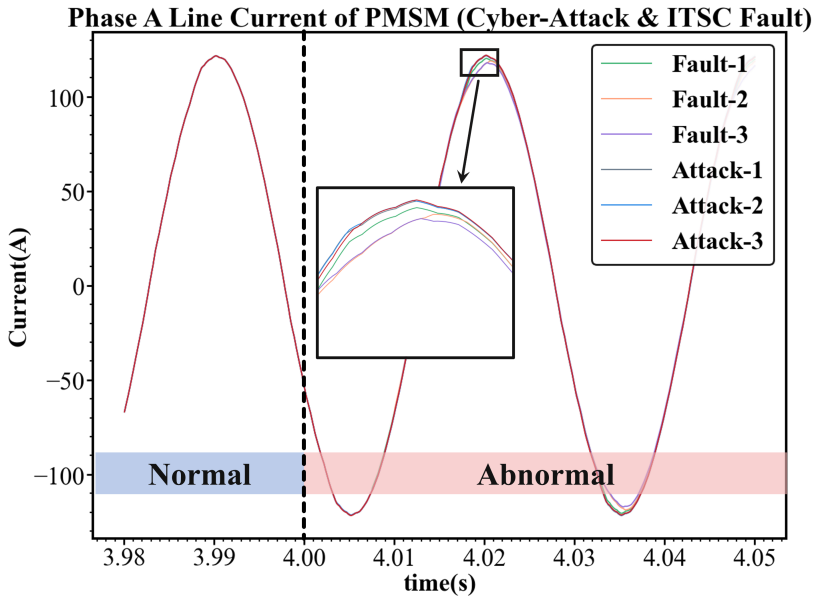


Figure 4.24: PMSM line current waveforms in different ITSC faults and FDI attacks scenarios.

The detector locates at the PCC, denoted as a grey dashed circle in fig. 4.21 and extracts the PCC line current signals.

In this case study, the fault scenarios include the inter-turn short circuit faults in PMSM stator windings and bearing faults in the induction machine. Different fault scenarios are designed by setting different

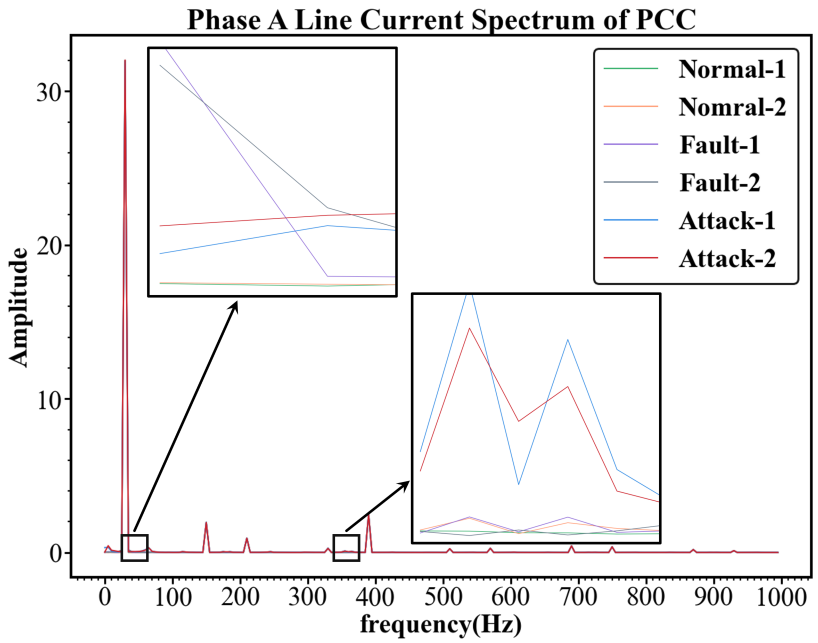


Figure 4.25: Spectrum of PCC line currents in different fault and attack scenarios.

fault parameters in the fault models described in section II. Meanwhile, the cyber-attack scenarios are also designed using the adversary models described in section II. According to the adversary model, the attacks are dependent on the available resources to the attackers. Usually, the more resources they have, the more sophisticated the attack will be. On the other hand, naive attacks with limited adversary resources are more likely to cause drastic impacts on the systems. Such naive attacks are comparably easier to detect because the attack impacts are more significant and noticeable to the detectors. However, the impacts will be more subtle and much more challenging to detect for more sophisticated attacks. For these more sophisticated attacks, traditional detectors will not be sufficient. Therefore, the attack scenarios considered in this section are specifically designed to achieve the objectives established by the attackers. More specifically, these attacks will try to emulate some of the fault responses and trigger the false alarms to the original system monitors. Recently, as many applications have adopted physical fault detectors based on current signature analysis, the detection results heavily depend on specific frequency components. Therefore, these attacks could confuse the original fault detectors and trigger false alarms, leading to unnecessary shutdown and maintenance. For example, fig. 4.23 and fig. 4.24 show some waveforms of IM and PMSM in different fault and such types of attack scenarios. In these scenarios, the attackers use FDI attacks on the motor drive controller current feedbacks and voltage command outputs to mimic the bearing and ITSC faults in IM and PMSM, respectively. As suggested by these waveforms, the faults and attacks are highly similar. However, these attacks are not identical and leave traces in the PCC line currents. Fig. 4.25 shows some PCC line current spectra from the above scenarios. According to the spectra, there are still some

distinguishable signatures, but it will be challenging for traditional current signature analysis because the signatures are too subtle. Therefore, it is beneficial to adopt the proposed data-driven detection method in these situations.

In summary, the case study in this section includes 128 scenarios covering different bearing faults, ITSC faults, FDI attacks, and operating conditions, and, with these scenarios, we generated 12800 samples of the PCC line current measurements. Among these samples, 80% are randomly selected as the training data sets, and the rest 20% are the testing sets.

Table 4.14: Confusion Matrix of Testing Reuslts (Normalized)
(Overall detection accuracy: 95.5%)

		Prediction		
		Attack	Normal	Fault
Reference	Attack	0.96	0.01	0.03
	Normal	0	1	0
	Fault	0.04	0.01	0.95

Table 4.14 shows the overall testing results, and fig. 4.26 shows the detection results of individual classification methods. According to the results, the individual classifiers tend to have poor performances as predicted, and the majority vote procedure is proved promising.

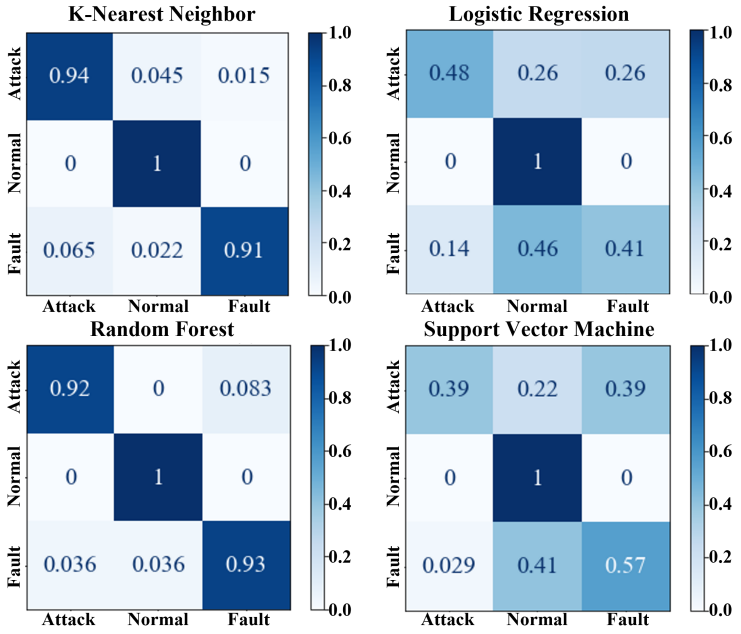


Figure 4.26: Confusion matrices of the detection accuracy from individual classification methods (Normalized).

4.4 Data-Driven Cyberattack Detection for Intelligent Motor Drives with Limited Experimental Data

In the past decade, concerns about cybersecurity in intelligent power electronics systems have risen due to pervasive implementations of networked digital control units. (Yang, Guo, Li, et al., 2020b; J. Ye et al., 2022; J. Ye et al., 2021; D. Zhang et al., 2021) demonstrated vulnerabilities and impacts of modern power electronics systems from a wide range of applications, such as photovoltaic (PV) systems, electric vehicles, and intelligent manufacturing systems. Recent research has focused on different detection approaches targeting diverse power electronics applications, such as DC microgrids, PV farms, and industrial motor drives, to address these concerns and enhance the reliability of intelligent power electronics systems. Most recent proposed detection methods could be categorized as physics-based methods or data-driven methods.

Common physics-based methods detect cyber-attacks by analyzing pre-defined system performance metrics or residuals between predicted system variables and corresponding true measurements(Giraldo et al., 2018). For example, (Sahoo et al., 2019) proposed a cooperative vulnerability factor for each power electronics agent within the microgrid to detect stealthy cyber-attacks. (Gupta et al., 2022) characterized the local voltage and frequency measurements to a 2D feature space so that the characterized metrics could distinguish cyber-attacks and physical faults in microgrids. (Beg et al., 2021) used the signal temporal and time-frequency logic formalism to detect anomalies in microgrids. (Sahoo et al., 2020) developed a detection approach targeting microgrid false-data-injection attacks using the discordant element. (J. Zhang et al., 2022) developed an attack detector for PV farms based on the harmonic state space models. (J. Zhang et al., 2021) proposed a residual-based detector for island microgrids using the harmonics state-space matrix and space phase model. (Yang, Guo, & Ye, 2021b) proposed an anomaly detection method for motor drives in electric vehicle powertrains using self-defined frequency-domain metrics. However, most physics-based methods rely on accurate physical models of the target systems, which are not available for most cyber-attack scenarios. In real-world applications, cyber-attacks are highly unpredictable, and their analytical impact models depend highly on specific attack policies. These factors will make the performances of most physics-based methods unreliable.

Recent research started to harvest the power of data-driven methods to develop model-free detection methods in power electronics systems to reduce the dependency on physical models. (Habibi et al., 2021) adopted a particular type of recurrent neural network, namely a nonlinear auto-regressive exogenous model, to detect false data injection attacks in microgrids. (Dehghani et al., 2021) proposed an attack detection method by combining deep neural network and wavelet singular value decomposition. (Khan et al., 2021) used multi-class support vector machines to detect and localize false-data-injection and denial-of-service attacks in inverter-based systems. (F. Li et al., 2021) proposed a detection and diagnosis method targeting data integrity attacks in solar farms using a multilayer long short-term memory network. (Q. Li et al., 2020) examined the effectiveness of different standard data-driven methods with micro-PMU data in detecting cyber-attacks in PV farms. (Yang & Ye, 2022; Yang, Ye, & Guo, 2022) developed anomaly detection methods for electric vehicle traction motor drives using a combined support vector machine,

random forests, k-nearest-neighborhood, and logistic regression. (Yang, Ye, Coshatt, et al., 2022) adopted supervised classification methods to distinguish cyber-attacks and physical faults in manufacturing motor drives.

Despite the advantages of recently developed data-driven approaches, a major obstacle in deep learning-based cyber-attack detection in power electronics systems is the requirement for large volume training data sets. The model can only learn features incorporated in the training data, and the algorithm may fail when testing data contains different features (Wen et al., 2019; Y. Zhang & Yan, 2019). To address this issue, a large-scale training dataset is necessary to incorporate similar data with testing data. However, the computational cost due to the large volume of training data hinders deep learning model performance. Transfer learning techniques have been proposed to enable machine learning models to leverage knowledge from one domain to another (Maschler & Weyrich, 2021), thus reducing the amount of required training data (Q. Li et al., 2022). Deep transfer learning methods have been utilized in cyber-attack detection and fault diagnosis in intelligent machine systems. Some methods (X. Li et al., 2020; P. Liao et al., 2022; Y. Zhang & Yan, 2019) employ deep adversarial models to achieve transfer learning by minimizing predicted domain labels, while others aim to minimize the discrepancy between learned features from the source and target domains (Wen et al., 2019). Nevertheless, most existing deep-learning-based methods for intelligent power electronics systems rely on simulation data rather than hardware experimental data sets. Compared to experimental data sets, models trained on simulation data are more challenging to implement in real-world environments, particularly for mission-critical power electronics systems. Generating large amounts of experimental data sets faces two challenges: (1) experimental data generation requires substantial resources, and (2) cyber-attack experiments on intelligent power electronics systems pose risks and may cause damage to the environment and human health. Consequently, acquiring vast amounts of experimental data for training data-driven models is unrealistic in practice.

This study aims to address this issue by combining simulation and experimental data sets using transfer learning, enabling the resulting motor drive attack detector to perform well even with limited experimental data. Fig. 4.22 shows a general flow chart of the proposed method. The method first trains a convolutional neural network (CNN) with extensive simulation data. Then it uses transfer learning to fine-tune the pre-trained CNN model with limited experimental data. In other words, the proposed method first learns the primary characteristics of different cyber-attacks with simulation data. Then it learns how to adapt the existing knowledge to real-world target prototype systems with limited experimental data. The contributions of the proposed method to existing data-driven attack detection research could be summarized as follows:

1. This work proposes a novel approach for cyber-attack detection in motor drives using Transfer Learning based on Convolutional Neural Networks (CNN).

2. The proposed method achieves outstanding detection performance with an accuracy of 99.5% while reducing development time, costs, and risks.

3. The proposed transfer learning model effectively reduces convergence time during training in the presence of limited experimental data.

4. The proposed model maintains satisfactory detection accuracy of over 96% even when experimental training data is highly limited (10% of available data).

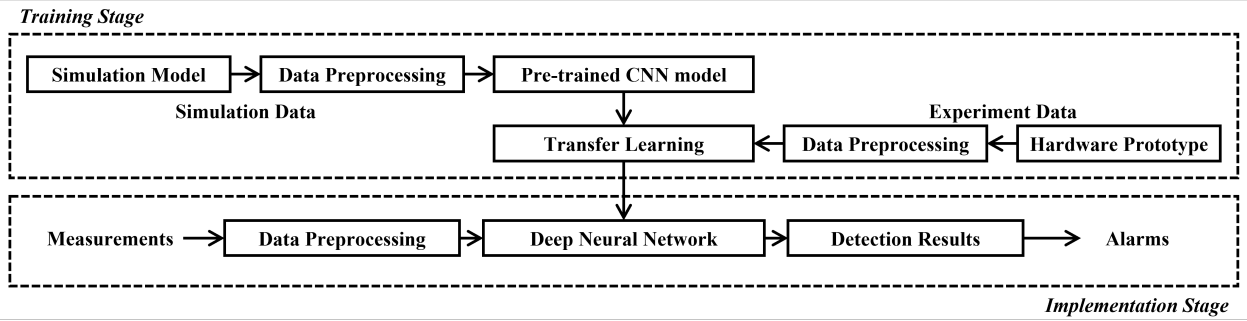


Figure 4.27: General flow chart of the proposed method.

4.4.1 The Proposed Detection Strategy

Existing methods for detecting anomalies in power electronics and electric machines include physics-based and data-driven approaches. Physics-based methods rely on the expertise of the target systems. Common physics-based methods include using residuals between measured and predicted system variables and self-defined system metrics to detect anomalies. Such methods have been widely used in fault detection for power electronics and electric machines in recent decades. However, with the rising concerns of cyber-attacks in recent years, physics-based methods exposed several weaknesses:

1. There is no accurate and fixed physical model for predicting system behaviors under cyber-attacks.
2. Residuals and metrics used in physics-based methods are inflexible, while cyber-attacks are constantly evolving and highly unpredictable. Therefore, such methods may not be effective in cyber-attack scenarios.

In recent years, with the emergence of data-driven methods for detecting cyber-attacks in power electronics and electric machines, the advantages of such approaches started to reveal:

1. Data-driven methods do not require an accurate physical model.
2. Data-driven methods can adapt to different systems without changing their original architecture.

However, the core requirements for all data-driven methods are the quality and quantity of the available training data sets. So far, the primary sources for such data sets in power electronics and electric machines are simulations and experiments. Each source has its strengths and weaknesses. On the one hand, simulations could generate a large amount of data with considerably low costs. However, they can only catch some of the details of target systems due to limited computational power. On the other hand, experiments

could accurately reflect system behaviors, but the costs and risks of cyber-attack experiments in power electronics and electric machines are substantial.

Therefore, this study proposed a new approach to develop the cyber-attack detector using transfer learning based on CNN to combine the advantages of both simulation and experiment data sets (Fig. 4.22). The proposed method first uses extensive simulation data sets to train a CNN model. This pre-trained model includes a convolutional network and a fully-connected network. The convolutional network will map the input data to a feature space. The fully-connected network will map this feature space to the estimated possibilities for each category. After acquiring this pre-trained CNN model based on simulation data, the proposed method will use the transfer learning method to fine-tune this pre-trained CNN model with limited experimental data. The rest of this section will elaborate on the details of the proposed method.

Training Stage

The training stage includes five tasks: generating simulation data; collecting experiment data; preprocessing collected data; pre-training the CNN model; and fine-tuning the CNN model using transfer learning.

Simulation Data Generation: The proposed method uses simulation data to pre-train the CNN model. Therefore, the simulation data needs to reflect the significant impacts of cyber-attacks on motor drives and avoid trivial details like switching-caused ripples and measurement noise to lower the computational costs. Fig. 4.28 shows the adopted simulation model based on the motor drive average model and kill-chain-liked control information flow (CIF) model. Such a simulation model first maps different cyber-attacks to the CIF model, which reflects how attacks propagate in the motor drive controller. Fig. 4.29 shows an example of the CIF model for a permanent magnet synchronous motor (PMSM) drive with field-oriented control. The red paths in Fig. 4.29 denote the propagation paths of a false data injection attack targeting one of the offset variables for analog-to-digital-conversion (ADC) units. The resulting tainted control law for this example is shown in eq. (4.21) - eq. (4.28), where $z_m,\ z_d,\ z_q$ are the controller state variables, $i_d,\ i_q,\ \omega_m$ are the current and speed feedback variables, $\Omega_m,\ I_d,\ I_q$ are the references, $v_d,\ v_q,\ u_d,\ u_q$ are the controller output variables, θ is the rotor position angle, L_s is the machine winding inductance, λ_{PM} is the flux linkage of the permanent magnet, k_{adc} is the ADC gain of the MCUs, K_x are controller parameters, and α is the attack coefficient.

The simulation model will then map the tainted control laws to the average model for the physical plant (i.e., motor drive) and solve the tainted dynamic equations to get system state trajectories under attack. Finally, such system state trajectories will be converted to measurement samples and organized as data sets for later training processes. Fig. 4.31 shows one sample of the measurement (motor line currents)

waveform from simulation data sets.

$$\frac{dz_m}{dt} = \Omega_m - \omega_m \tag{4.21}$$

$$\frac{d\hat{z}_d}{dt} = I_d - (i_d - \frac{2}{3}k_{adc} \cdot \cos\theta \cdot \alpha) \tag{4.22}$$

$$\frac{d\hat{z}_q}{dt} = I_q - (i_q + \frac{2}{3}k_{adc} \cdot \sin\theta \cdot \alpha) \tag{4.23}$$

$$I_q = K_{mp}(\Omega_m - \omega_m) + K_{mi} \cdot z_m \tag{4.24}$$

$$\hat{u}_d = K_{dp}(I_d - (i_d - \frac{2}{3}k_{adc} \cdot \cos\theta \cdot \alpha)) - K_{di} \cdot \hat{z}_d \tag{4.25}$$

$$\hat{u}_q = K_{qp}(I_q - (i_q + \frac{2}{3}k_{adc} \cdot \sin\theta \cdot \alpha)) + K_{qi} \cdot \hat{z}_q \tag{4.26}$$

$$\hat{v}_d = \hat{u}_d - \omega_m L_s(i_q + \frac{2}{3}k_{adc} \cdot \sin\theta \cdot \alpha) \tag{4.27}$$

$$\hat{v}_q = \hat{u}_q + \omega_m L_s(i_d - \frac{2}{3}k_{adc} \cdot \cos\theta \cdot \alpha) + \omega_m \lambda_{PM} \tag{4.28}$$

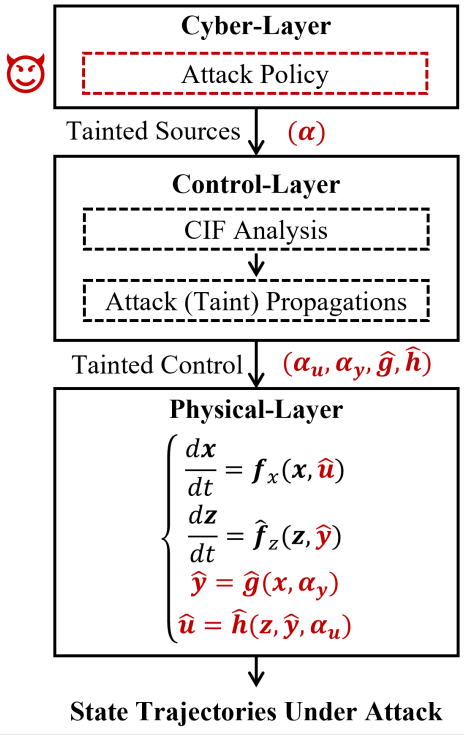


Figure 4.28: Diagram of the simulation model, where α_u, α_y, $\hat{g}$, $\hat{h}$ are the attack coefficients and tainted control laws, $\hat{y}$, $\hat{u}$ are the resulted tainted feedback variables and control commands, and x, z, f_x, f_z are the state variables and equations for physical plants and digital controllers.

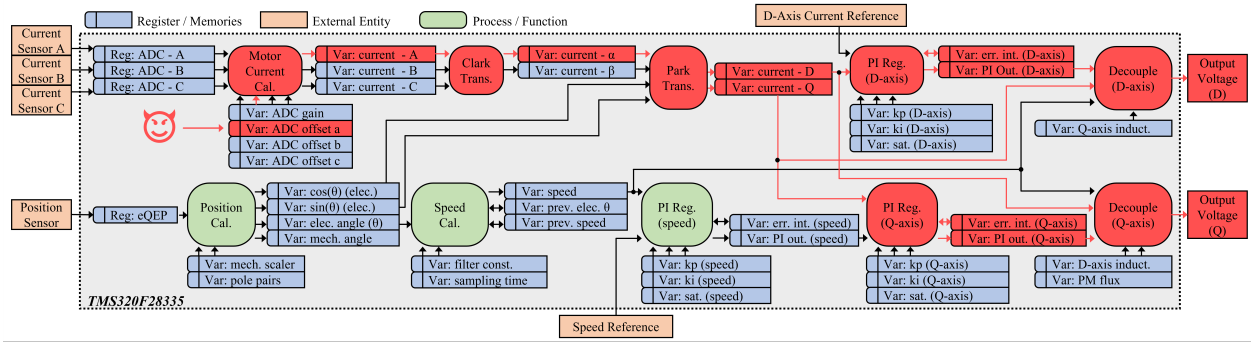


Figure 4.29: An example of the kill-chain-liked control information flow model.

Experiment Data Collection: The proposed method uses experimental data to fine-tune the pre-trained CNN model so that the final model can perform well in real-world target motor drives. As the pre-trained model covers most structures and features of different cyber-attacks, the experiment does not need to include many attack scenarios. Fig. 4.30 shows a diagram of the prototype motor drives used to generate experiment data sets. Such a prototype implements the control algorithms in a digital signal processor (DSP) connected to a host computer. The host computer is exposed to public networks where the malicious hacker is trying to attack the prototype motor drive. As experimented cyber-attack scenarios need to be fully controllable, such prototype implants pre-defined malicious codes to the DSP and leaves some 'back-doors' to trigger these codes. Therefore, the hacker will explore these 'back-doors' and trigger the pre-defined cyber-attacks, reducing the risks of unpredictable impacts during experiments. Fig. 4.31 shows a sample waveform from experiment data sets. It shows that experiment data sets add real-world ripples and noises while the impact of cyber-attacks is similar to the simulation results.

Data Preprocessing: The primary objective of the proposed methodology is to enable continuous monitoring of the target system. Consequently, the method involves the periodic collection of a fixed quantity of samples (window size) from sensor signals (e.g., motor line currents) at a predetermined sampling frequency. Upon obtaining the raw measurement samples, the method transforms these samples into a specific format suitable for the input layer of the Convolutional Neural Network (CNN) model. Given that the majority of cyber-attacks targeting motor drives tend to induce undesired harmonics in motor line currents, these attacks become more discernible once the motor line currents are transferred to their frequency domains. As such, the proposed methodology initially employs the Fast Fourier Transform (FFT) to convert the sampled three-phase motor line currents into three distinct spectra. In this study, a sampling rate of 25 kHz and a window size of 1250 points (0.05 s) are used. Furthermore, since most harmonics resulting from cyber-attacks are concentrated in the lower frequency range, the first 100 elements (0 Hz to 2000 Hz) are chosen to reduce the size of input features.

Pre-training Process with Simulation Data: Utilizing the developed simulation model, it is possible to generate a substantial volume of data encompassing diverse attack scenarios and operating conditions.

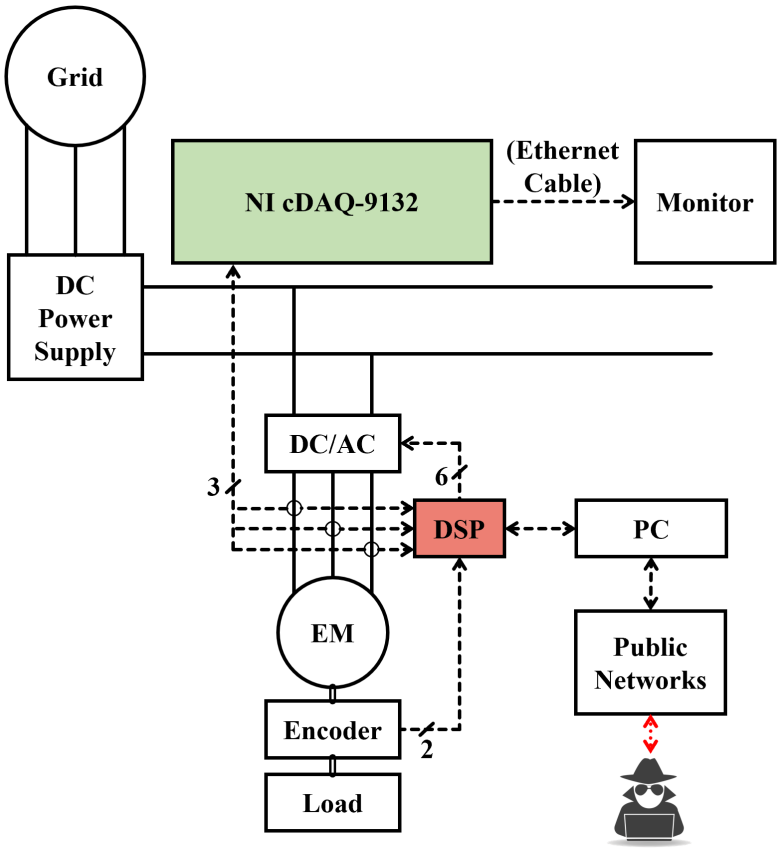


Figure 4.30: Diagram of the prototype motor drives used to generate experiment data sets.

Since the preprocessing step employs the FFT to convert raw measurements into three sets of motor line current signature spectra, the input layer of the CNN model accepts three channels of 1-D arrays. Subsequently, the model classifies these spectra into normal conditions and various attack scenarios. Table 4.15 and fig. 4.32 show the proposed method's general structure of the CNN model. As shown in table 4.15, layer 0-9 is the backbone of the CNN model containing convolutional layers (Conv1d), nonlinear activation functions (ReLU), and maximum pooling layers (MaxPool1d). This first part of the CNN maps the input spectra images to a 10×128 feature space. Such feature space captures the primary structures of normal operating conditions and different cyber-attack scenarios. Then, layer 10 flattens the feature space to a 1×1280 array. Such an array is mapped to 3 possibilities corresponding to each scenario. (The example used in this study includes two types of cyber-attack scenarios. Therefore, there are three possible classification outputs: normal condition, attack scenario 1, and attack scenario 2.) As shown in table 4.15, there are 203,651 trainable parameters in this CNN model. These parameters are trained and validated by a large amount of simulation data from multiple scenarios.

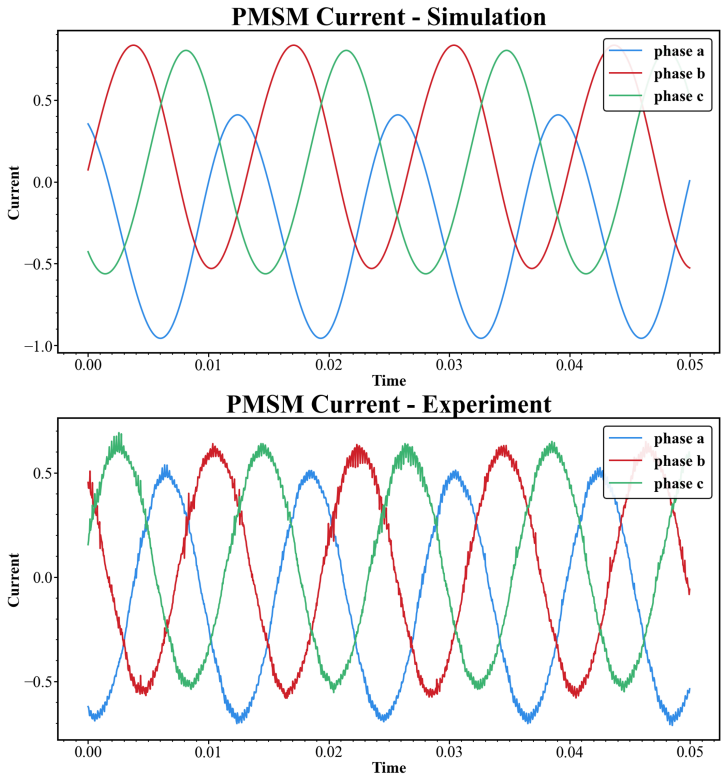


Figure 4.31: Samples of simulation (top) and experiment (bottom) data sets.

Transfer Learning with Experiment Data: As indicated in Fig. 4.31, the primary patterns of cyber-attacks exhibit similarities between the simulated and experimental waveforms. Consequently, the objective of transfer learning is to incorporate subtle details (such as switching-induced ripples, mechanical disturbances, and measurement noise) into the pre-trained model, enabling the final model to accurately detect and distinguish normal conditions and various cyber-attack scenarios. Specifically, the proposed method freezes the parameters from layers 0-10 and sets the pre-trained parameters of layers 11-13 as initial values. The network is then re-trained with a limited amount of experimental data. Due to the well-developed pre-trained model, the re-training process requires only a minimal amount of experimental data. Simultaneously, as shown in Table 4.15, the number of trainable parameters in the transfer learning model is reduced from 203,651 to 172,419, further decreasing the training time for transfer learning. Upon completing transfer learning, the resulting classifier accepts preprocessed signature spectra as inputs, outputs probabilities for each scenario, and generates detection results by selecting the scenario with the highest probability.

Table 4.15: CNN Structure

Layer (type: index)	Output Shape	Number of Parameters
Input: 0	[:,100,3]	–
Conv1d: 1	[:,98,32]	320
ReLU: 2	[:,98,32]	–
MaxPool1d: 3	[:,49,32]	–
Conv1d: 4	[:,47,64]	6,208
ReLU: 5	[:,47,64]	–
MaxPool1d: 6	[:,23,64]	–
Conv1d: 7	[:,21,128]	24,704
ReLU: 8	[:,21,128]	–
MaxPool1d: 9	[:,10,128]	–
Flatten: 10	[:,1280]	–
Linear: 11	[:,128]	163,968
Linear: 12	[:,64]	8256
Linear: 13	[:,3]	195

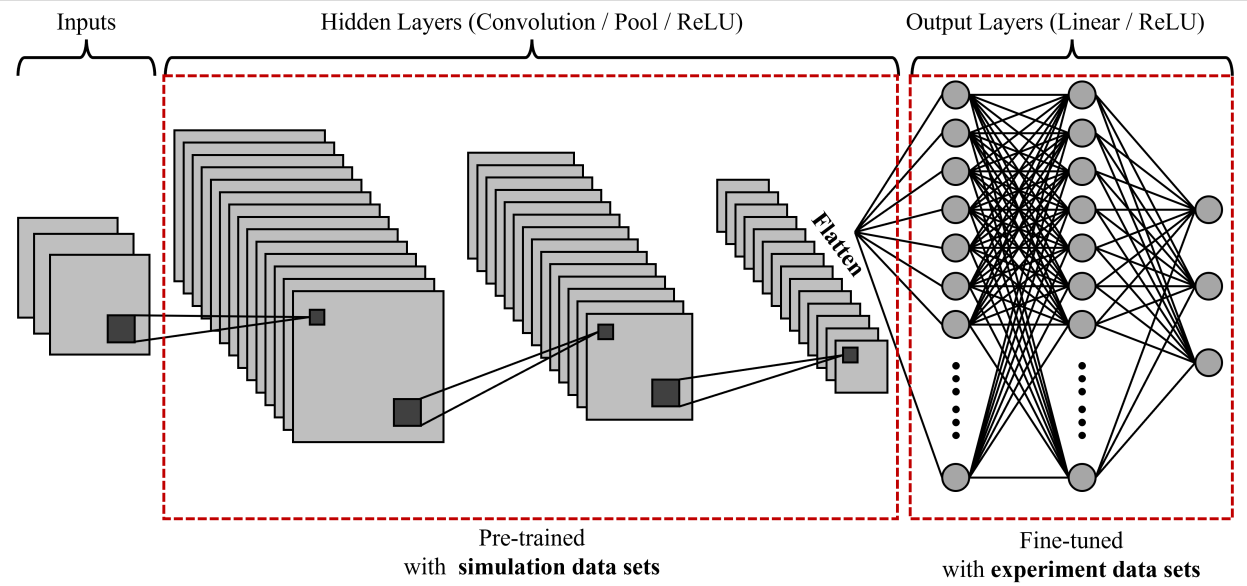


Figure 4.32: Diagram of the CNN structure.

Implementation Stage

After generating the final classification model in the training stage, the implementation stage will continuously use the developed classifier to monitor the target motor drives. The online monitoring system will follow a straightforward logic shown below:

- Step 1: Sample the measured current sensor signals with specified window size and sampling frequency.
- Step 2: Convert the raw measurement to motor line current signature spectra using the same parameters in the training stage.
- Step 3: Feed the spectra to the final classification model from the training stage.
- Step 4: Generate monitoring results based on the classifier output.
- Step 5: Update detection result and clear the workspace except for the preprocessing parameters and classification model.
- Step 6: Start a new monitoring cycle from Step 1.

4.4.2 Simulation And Experiment Case Study

This section forms a case study including two types of cyber-attacks targeting a 1.5 kW PMSM motor drive to test and validate the proposed method. The rest of this section describes details of the case study and simulation and experiment setups.

Case Study

The case study in this section includes typical false-data-injection (FDI)attacks targeting two control variables: the offset variable of the ADC unit and the calculated speed feedback variable.

Scenario 1: FDI attack on the motor current offset variable: In practical motor drive controllers, current offset variables are critical to compensate for the current sensors' zero drift issues. Most controllers calculate such offset variables during the initialization process. After initialization, these offsets will maintain constant. However, as these offsets are stored in the memory data sections, multiple attacks could access these variables and maliciously modify them, such as buffer-overflow attacks and FDI attacks. This case study considers a scenario where the motor phase A current sensor offset variable is under an FDI attack. According to the CIF model in fig. 4.28, the tainted variable and attack policy is shown in eq. (4.29),

$$\hat{x}_{offsetA} = x_{offsetA} + \alpha \tag{4.29}$$

where $x_{offsetA}$ and $\hat{x}_{offsetA}$ is the original and attacked motor phase A current offset variables; α is the attack coefficient. The resulting tainted control laws for this scenario is shown in eq. (4.21) - eq. (4.28).

Scenario 2: FDI attack on the motor speed feedback variable: Besides current offset variables, the calculated speed feedback is also a vulnerable target of malicious attacks. For example, the Stuxnet worm compromised the industrial control system by manipulating the rotating speeds of industrial motor drives. Suppose the attack policy is the same as eq. (4.29), which is shown in eq. (4.30), where ω_m and $\hat{\omega}_m$ is the

original and attacked motor speed feedback variables.

$$\hat{\omega}_m = \omega_m + \alpha \tag{4.30}$$

Then, the resulting tainted control law is shown in eq. (4.31) - eq. (4.38).

$$\frac{d\hat{z}_m}{dt} = \Omega_m - \hat{\omega}_m = \Omega_m - (\omega_m + \alpha) \tag{4.31}$$

$$\frac{dz_d}{dt} = I_d - i_d \tag{4.32}$$

$$\frac{d\hat{z}_q}{dt} = \hat{I}_q - i_q \tag{4.33}$$

$$\hat{I}_q = K_{mp}(\Omega_m - (\omega_m + \alpha)) + K_{mi} \cdot \hat{z}_m \tag{4.34}$$

$$u_d = K_{dp}(I_d - i_d) - K_{di} \cdot z_d \tag{4.35}$$

$$\hat{u}_q = K_{qp}(\hat{I}_q - i_q) + K_{qi} \cdot \hat{z}_q \tag{4.36}$$

$$\hat{v}_d = u_d - (\omega_m + \alpha)L_s i_q \tag{4.37}$$

$$\hat{v}_q = \hat{u}_q + (\omega_m + \alpha)L_s i_d + (\omega_m + \alpha)\lambda_{PM} \tag{4.38}$$

Simulation Setups

Based on the tainted control laws described by eq. (4.21) - eq. (4.28) and eq. (4.31) - eq. (4.38), the simulation encompasses various attack coefficients α for both scenarios. Simultaneously, the simulation also considers different operating conditions for the motor drives. Table 4.16 provides details of the operating parameters and attack coefficients. With these diverse operating conditions and attack scenarios, the simulation comprises 336 ($4 \times 3 \times 14 \times 2$) distinct scenarios. Subsequently, the simulation extracts 100 samples from each scenario, forming a simulation dataset consisting of 67,200 samples labeled as 'normal condition', 'attack type 1', and 'attack type 2'. Fig. 4.33 displays two samples from the simulation dataset.

Table 4.16: Parameters for operating conditions and attack coefficient

Operating Speed (rpm)	Load Torque (N*m)	Attack Coefficient
900	1.38	±0.3
1200	1.85	±0.275
1500	2.31	±0.25
1800		±0.225
		±0.2
		±0.175
		±0.15

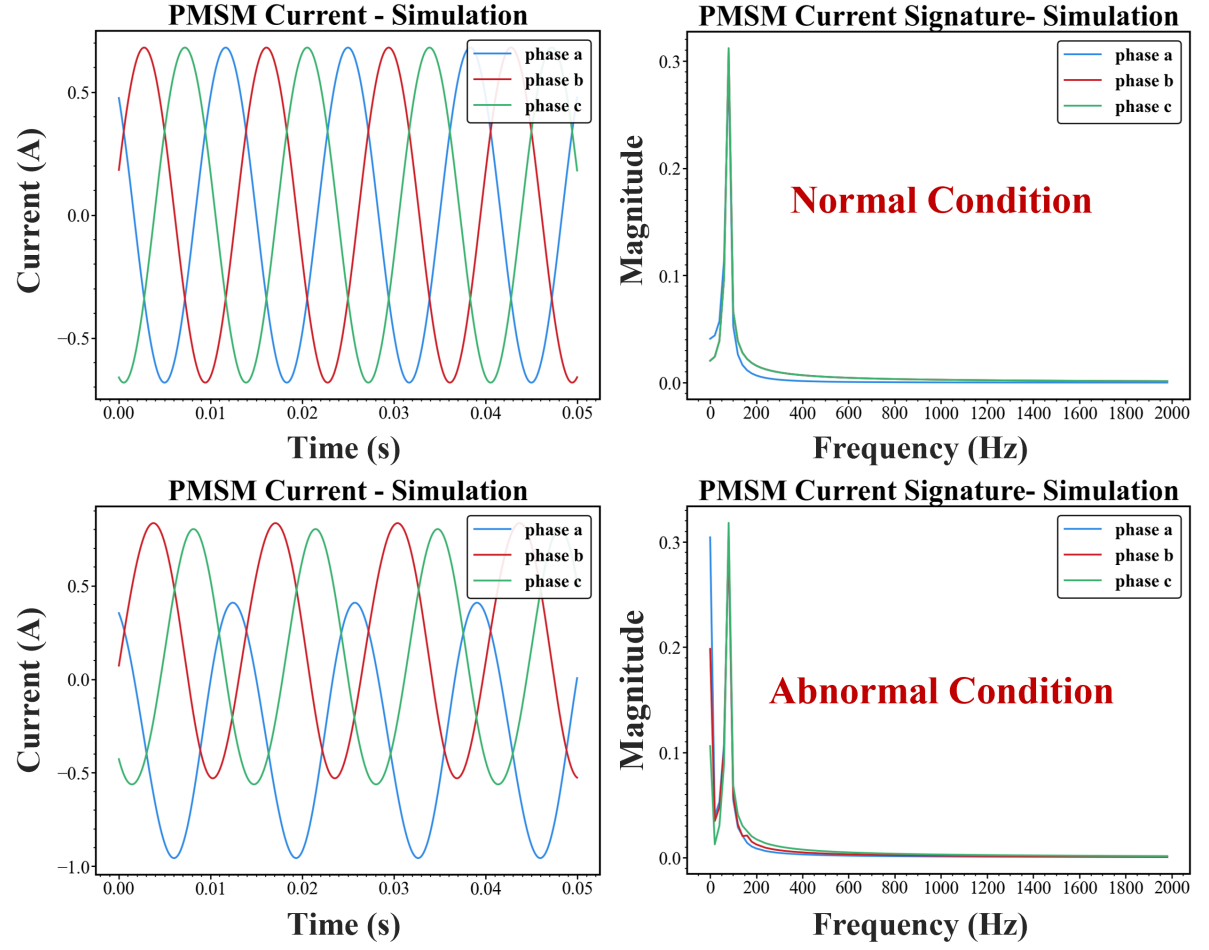


Figure 4.33: Sample plots of simulation data set.

Experiment Setups

Fig. 4.35 presents a photograph of the experimental prototype, while Table 4.17 provides a detailed overview of its specifications. The prototype features a 1.5 kW Permanent Magnet Synchronous Motor (PMSM) and shares the same structure illustrated in Fig. 4.30. Operating at a speed of 1000 rpm, the prototype experiences a mechanical load torque of approximately 2.77 N*m. Field-Oriented Control (FOC) algorithms, along with the malicious 'backdoor,' are implemented using a TMS320F28335 Microcontroller Unit (MCU) from Texas Instruments. Table 4.18 outlines the attack coefficients employed in various attack scenarios during the experiment. From the 10 distinct scenarios, a total of 2000 samples are extracted, effectively capturing the diverse characteristics of each scenario.Fig. 4.34 showcases two representative samples extracted from the experimental dataset. By incorporating these experimental samples into the training and validation process, the study aims to assess the proposed method's performance and gener-

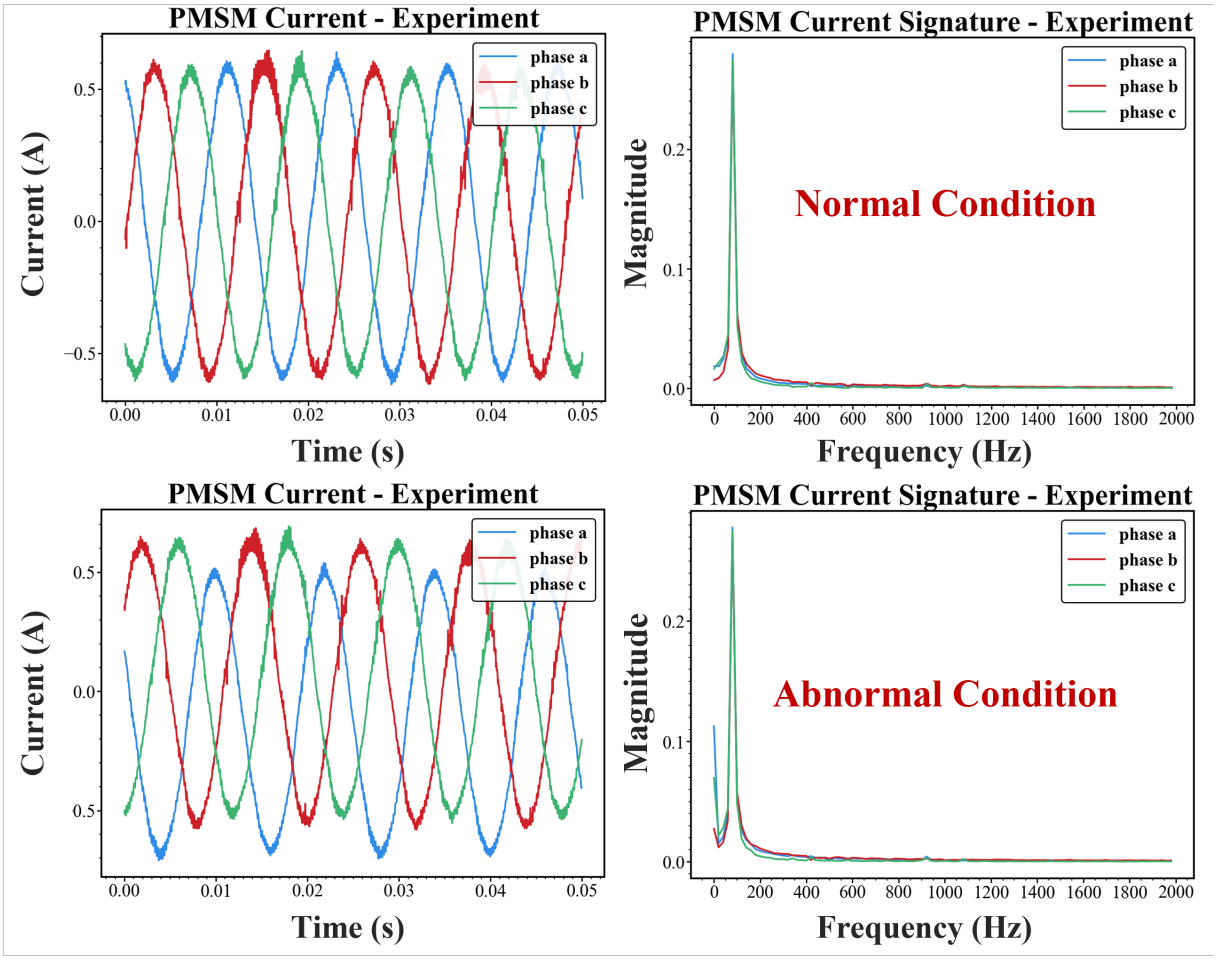


Figure 4.34: Sample plots of experiment data set.

alizability in real-world situations. Ultimately, this experimental setup serves to bridge the gap between simulation and real-world implementation, ensuring the development of a more reliable and robust cyber-attack detection system for motor drives.

Table 4.17: Specifications of the experiment platform.

Rated Power	1.5 kW	Stator Resistance	0.4050 Ω
Rated Current	8.2 A	Stator Inductance	0.0024 mH
DC Bus Voltage	200 V	Magnet Flux Linkage	0.0599 Wb
Rated Frequency	250 Hz	Number of Pole Pairs	5
Control Frequency	10 kHz	Motor Inertia	3.10e-4 kgm^2

Table 4.18: Parameters for attack coefficients in experiment

Scenario 1 (ADC Offset)	Scenario 2 (Speed Feedback)
0.1	0.05
0.2	0.1
0.3	0.15
-0.1	-0.05
-0.2	-0.1

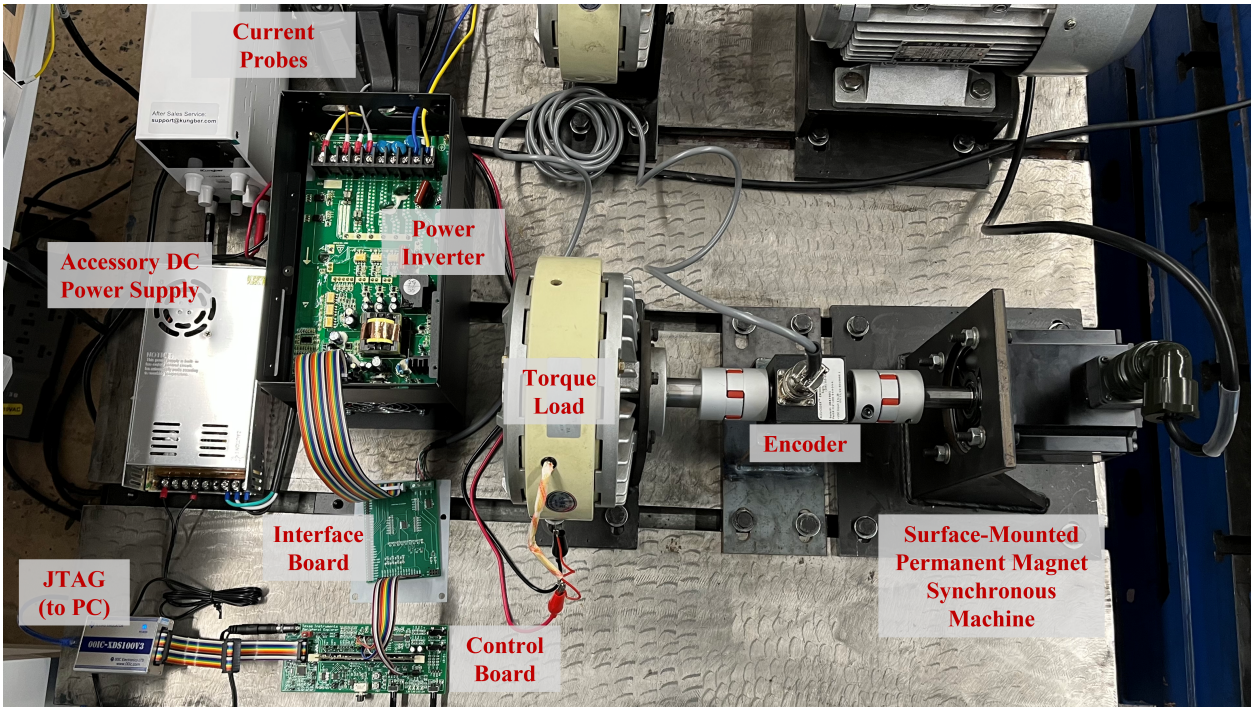


Figure 4.35: Picture of the hardware experiment platform with a PMSM drive.

4.4.3 Discussions on Validation Results

The simulation data (67,200 samples) and experimental data (2,000 samples) are initially divided into 80% for training data sets and 20% for validation data sets. Subsequently, the simulation data is employed to pre-train and validate the CNN model. The 80% experimental training data sets are considered as the overall available experimental data sets for training. Thereafter, only a fraction of these experimental training data (10%, 20%, 30%, 40%, 50%, 100%) is used to train transfer-learned CNN models, as well as new CNN models with identical structures from scratch. This section discusses the outcomes of the training and validation processes based on the aforementioned settings.

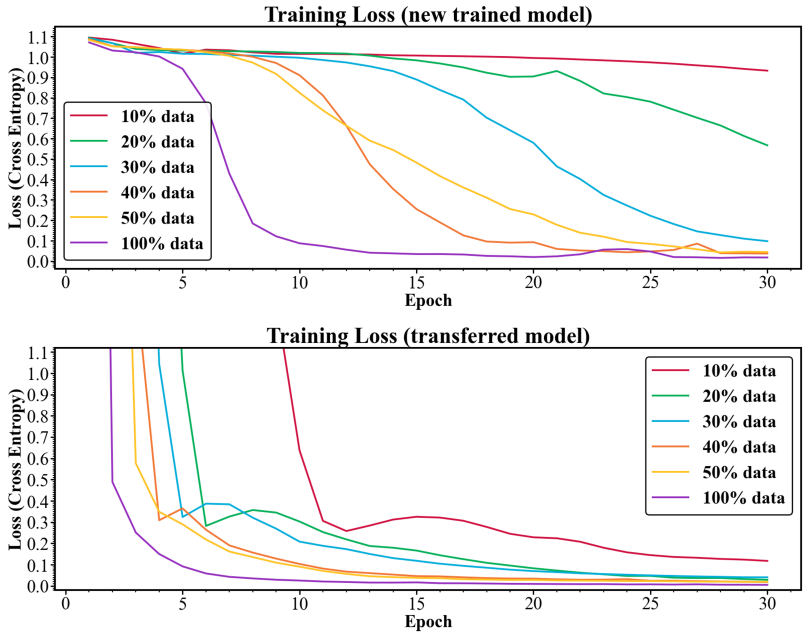


Figure 4.36: Comparison of the training loss for the transfer-learned CNN model and newly-trained CNN model, utilizing varying sizes of available experimental data sets.

Disscussion: Traning Loss

Fig. 4.36 presents the training losses for both transfer-learned CNN models and newly-trained CNN models, employing varying sizes of available experimental data sets. As indicated by Fig. 4.36, the initial losses for newly-trained models are smaller than those for transfer-learned models, suggesting that directly utilizing the parameters from the pre-trained model will yield higher losses compared to recalculating the initial parameters from available data sets. However, as the training process progresses, the losses of transfer-learned models rapidly converge to their minimum values, even when only 10% of the available experimental data sets are used. Conversely, the newly-trained models struggle to converge. Even with 100% of the available experimental training data, convergence still requires approximately 10 epochs, while transfer-learned models manage to converge after around 6 epochs. This finding supports the notion that the proposed transfer learning approach can reduce the convergence time of the training process when limited experimental data sets are available.

Discussion: Detection Accuracy

The detection accuracy is evaluated using experimental validation data sets, which are not involved in the training processes. Fig. 4.37 displays the confusion matrices, and Fig. 4.38 presents the overall accuracy for both transfer-learned CNN models and newly-trained CNN models, utilizing varying sizes of available

experimental training data sets. As indicated by these results, when sufficient experimental training data is available, the classification accuracy exceeds 99% for both transfer-learned and newly-trained models. However, when the available experimental training data is limited, the classification accuracy of newly-trained models declines significantly, whereas transfer-learned models maintain a satisfactory accuracy of above 96%. In Fig. 4.37, the newly-trained models struggle to detect any cyber-attacks with only 10% of available experimental training data, while the transfer-learned model achieves a notable 96.75% detection accuracy.

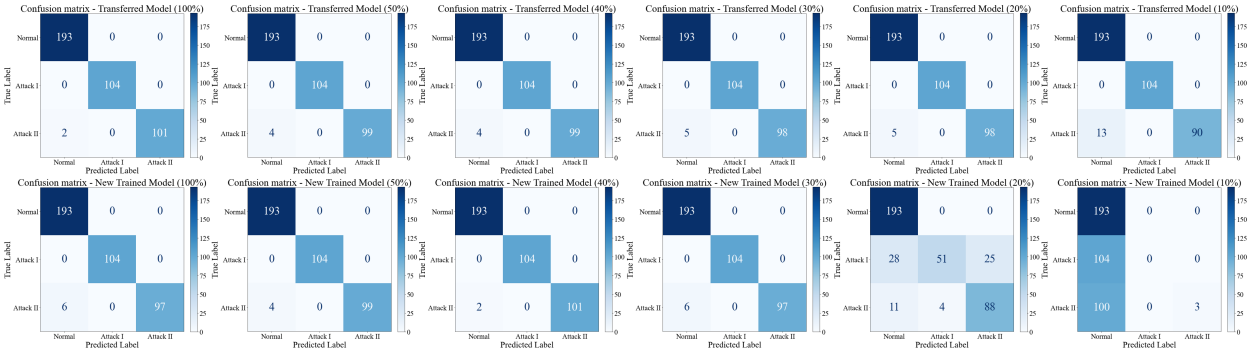


Figure 4.37: Confusion matrices from experimental validation sets for the transfer-learned CNN model and newly-trained CNN model, utilizing varying sizes of available experimental data sets.

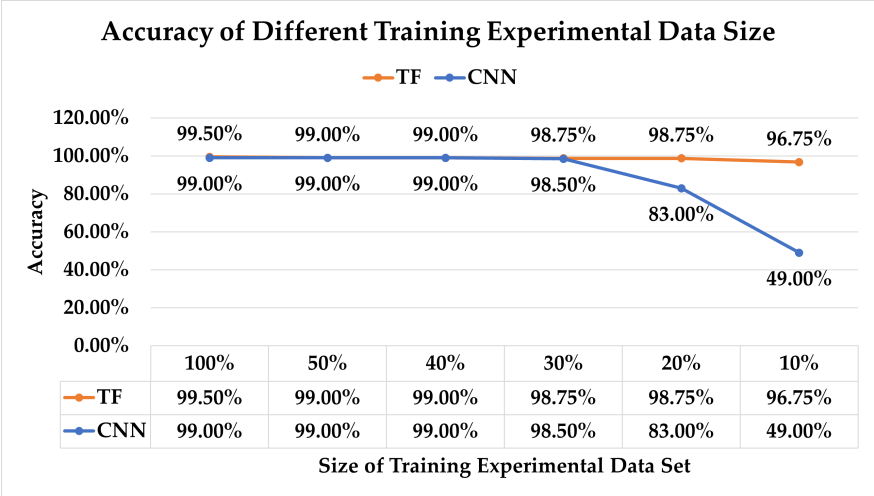


Figure 4.38: Overall classification accuracy for the transfer-learned CNN model and newly-trained CNN model, utilizing varying sizes of available experimental data sets. (TF: transfer-learned model, CNN: newly-trained CNN model)

4.4.4 Summary

In conclusion, this study presents a novel development approach for cyber-attack detection in motor drives, employing Convolutional Neural Networks (CNN) and transfer learning. The proposed method initially pre-trains a CNN model using extensive simulation data and subsequently fine-tunes the model through transfer learning with a limited amount of experimental data. This approach achieves outstanding detection performance, with an accuracy of 99.5%, while significantly reducing the development period, costs, and risks associated with high-performance monitoring systems for modern motor drives.

Our findings indicate that the initial losses for newly-trained models are smaller than those for transfer-learned models, suggesting that directly using parameters from the pre-trained model may result in higher losses compared to recalculating initial parameters from available data sets. However, as the training process advances, the losses of transfer-learned models swiftly converge to their minimum values, even with only 10% of available experimental data sets. In contrast, newly-trained models face difficulties in convergence. It takes approximately 10 epochs for them to converge when using 100% of available experimental training data, while transfer-learned models converge after around 6 epochs. These results support the notion that the proposed transfer learning approach can effectively reduce the convergence time of the training process in the presence of limited experimental data sets.

When assessing detection accuracy with experimental validation data sets not used during training, our results show that both transfer-learned and newly-trained models achieve classification accuracy exceeding 99% when sufficient experimental training data is available. However, when experimental training data is limited, the classification accuracy of newly-trained models declines considerably, whereas transfer-learned models maintain a satisfactory accuracy of over 96%. With only 10% of available experimental training data, newly-trained models struggle to detect any cyber-attacks, while the transfer-learned model achieves a remarkable 96.75% detection accuracy.

The substantial performance demonstrated by the proposed method contributes to the motor drive cyber-attack detection framework in several ways. It substantially reduces the reliance on large quantities of experimental data sets during the development process, lowers the costs and risks associated with cyber-attack detector development, strengthens the connections between simulations and experiments, and significantly shortens the development period by utilizing powerful simulation models.

Chapter 5

Demonstration Prototype: Security and Safety Monitoring Platform for Intelligent Electric Drive Systems

This chapter summarizes this dissertation's work by showing a demo prototype condition monitoring system for intelligent electric drive systems. Fig. 5.1 shows a diagram of this demo prototype system, which has the same structures as the one discussed in chapter 3. In addition, this prototype system implements detection and diagnostic algorithms developed in chapter 4. Meanwhile, this demo forms 14 attack scenarios targeting PMSMs and IMs. Tabel 5.1 shows the details of these attack scenarios.

These scenarios include four target onboard control-related resources. Three are the ADC offset variables for three-phase current feedbacks, respectively. The rest is the rotating speed reference variables. For ADC offsets, the attack falsely injects 0.05 bias into their original values. For speed references, the attack falsely injects a periodic disturbance to the original values. Such a disturbance has a magnitude of $\pm$ 0.01 p.u. and a period of 0.1s.

Furthermore, the developed prototype includes a real-time data visualization platform, showing the system conditions and some critical features. Fig. 5.2 shows a screenshot of this visualization dashboard when ACIM-1 is under attack 8 in 5.1.

This visualization dashboard is constructed with InfluxDB online cloud service. The monitoring system will first acquire real-time measurements from cDAQ-9132. Then, the features and detection results will be calculated from previously developed algorithms. In the end, the results will be sent to the InfluxDB cloud server. The dashboard shown in Fig. 5.2 will periodically visualize the data received in the cloud database. Fig. 5.3 shows the detailed flowchart for the monitoring algorithms implemented in the demo prototype.

As illustrated in the flowchart presented in Figure 5.3, five distinct classification models (Random Forests (RF), Multivariate Logistic Regression (LG), Support Vector Machines (SVM), K Nearest Neigh-

Table 5.1: Details of cyber-attack scenarios for demo prototype.

Case No.	Target System	Target Variables
1	PMSM	ADC offset - phase A current feedback
2	PMSM	ADC offset - phase B current feedback
3	PMSM	ADC offset - phase C current feedback
4	PMSM	ADC offset - phase A&B current feedback
5	PMSM	ADC offset - phase A&C current feedback
6	PMSM	ADC offset - phase B&C current feedback
7	PMSM	Speed reference
8	IM	ADC offset - phase A current feedback
9	IM	ADC offset - phase B current feedback
10	IM	ADC offset - phase C current feedback
11	IM	ADC offset - phase A&B current feedback
12	IM	ADC offset - phase A&C current feedback
13	IM	ADC offset - phase B&C current feedback
14	IM	Speed reference

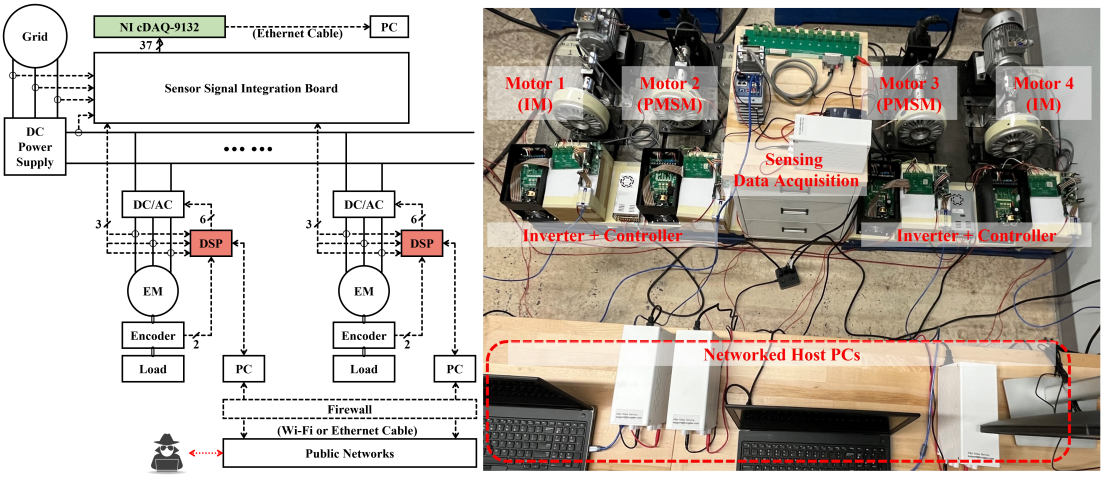


Figure 5.1: Diagram of the demo prototype condition monitoring system for intelligent electric drive systems.

bor (KNN), Convolutional Neural Networks (CNN)) have been evaluated. A sampling frequency of 2 kHz is utilized, with each monitoring window gathering 500 data points. The primary features extracted for detection and diagnostic purposes consist of the signal magnitude within the frequency domain. The testing process, as outlined by the scenarios in Table 5.1, comprises two levels for all monitors: individual monitoring of permanent magnet synchronous machine (PMSM) three-phase line currents, individual monitoring of induction machine (ACIM) three-phase line currents, and system monitoring of DC bus

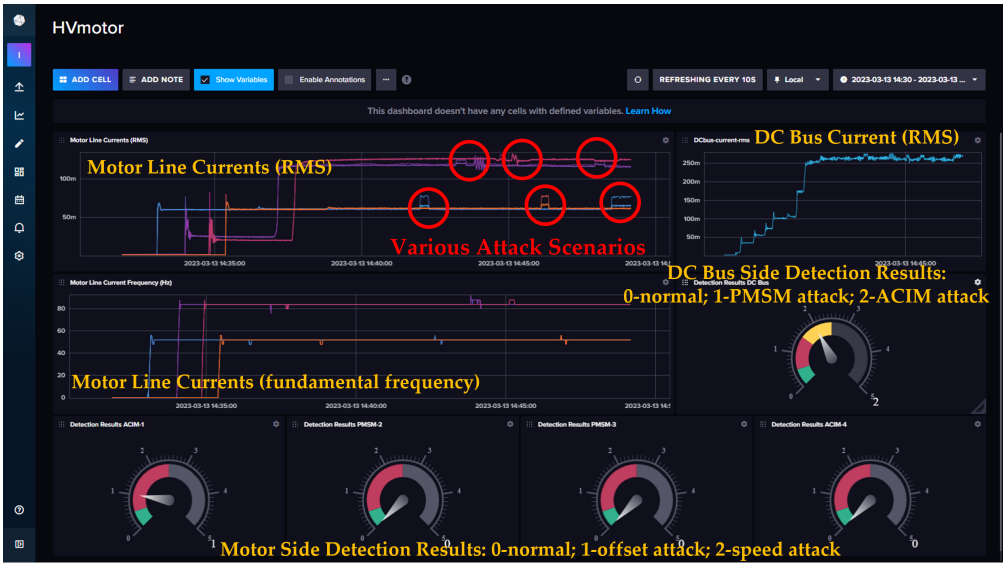


Figure 5.2: Screenshot of the real-time visualization dashboard when ACIM-1 is under attack 8 in 5.1.

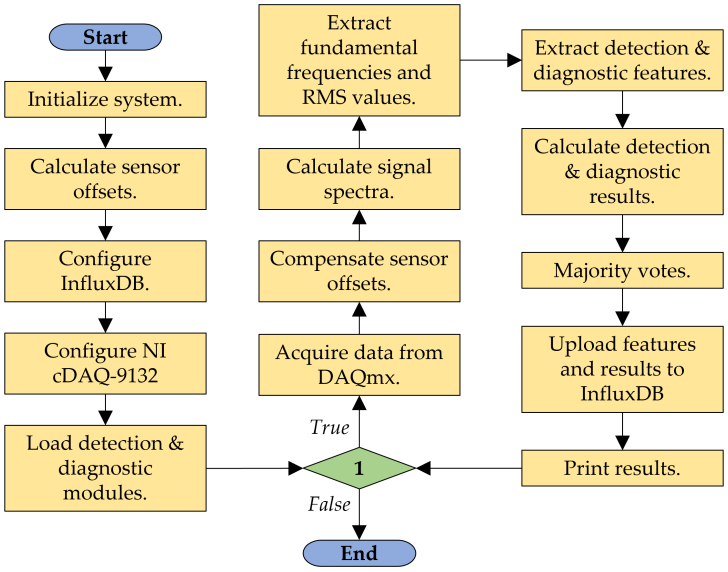


Figure 5.3: Flowchart of the demo prototype condition monitoring system software.

line current. For individual monitors, the first level aims to differentiate between normal conditions, ADC offset attacks (cases 1-6 and 8-13), and speed reference attacks (cases 7 and 14). The second level seeks to distinguish each attack scenario for both PMSM (cases 1-7) and ACIM (cases 8-14) monitors. Subsequently, in the case of the DC bus system monitor, the first level involves identifying normal conditions,

PMSM attacks (cases 1-7), and ACIM attacks (cases 8-14). The second level focuses on differentiating ADC offset attacks in PMSM (cases 1-6), speed reference attacks in PMSM (case 7), ADC offset attacks in ACIM (cases 8-13), and speed reference attacks in ACIM (case 14). Furthermore, an additional third level of complexity is introduced for the system monitor. This level entails distinguishing all 14 cases and normal conditions solely based on the DC bus line current. The following pages show the detailed results and some sample raw waveforms from different scenarios.

PMSM Individual Monitor Level I (3 class)

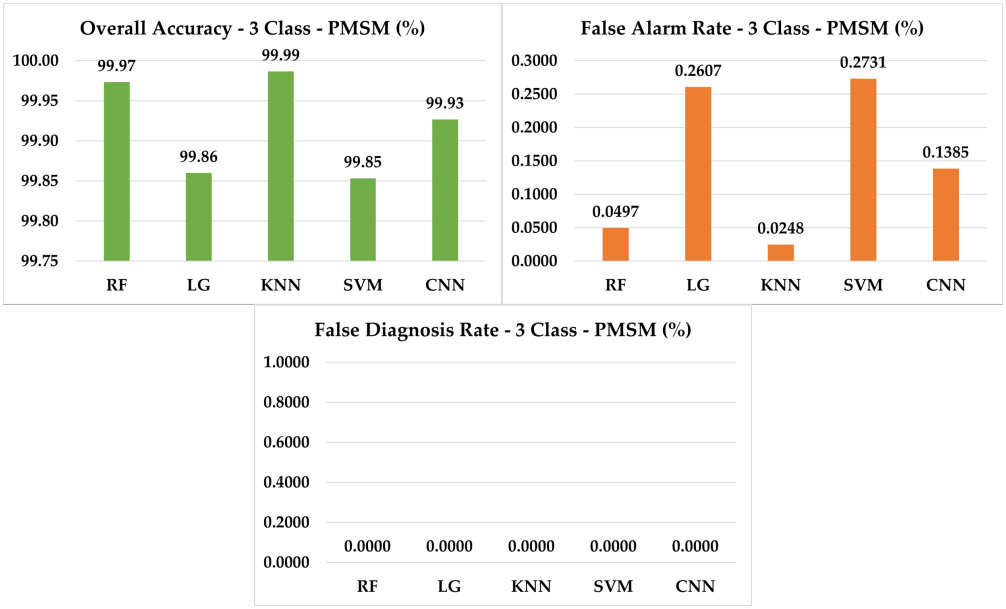


Figure 5.4: Overall accuracy, false alarm rate, and false diagnosis rate for 3-class PMSM individual monitor.

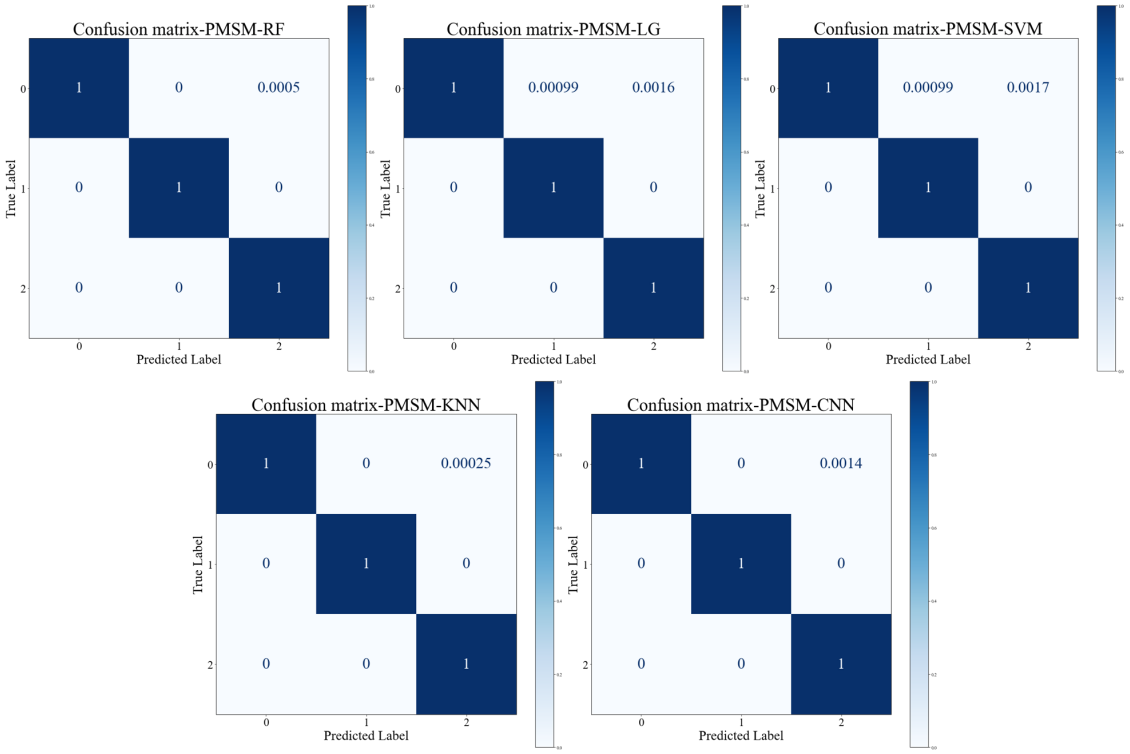


Figure 5.5: Confusion matrices for 3-class PMSM individual monitor. (0: normal condition, 1: ADC offset attacks, 2: speed reference attacks)

PMSM Individual Monitor Level II (8 class)

Figure 5.6: Overall accuracy, false alarm rate, and false diagnosis rate for 8-class PMSM individual monitor.

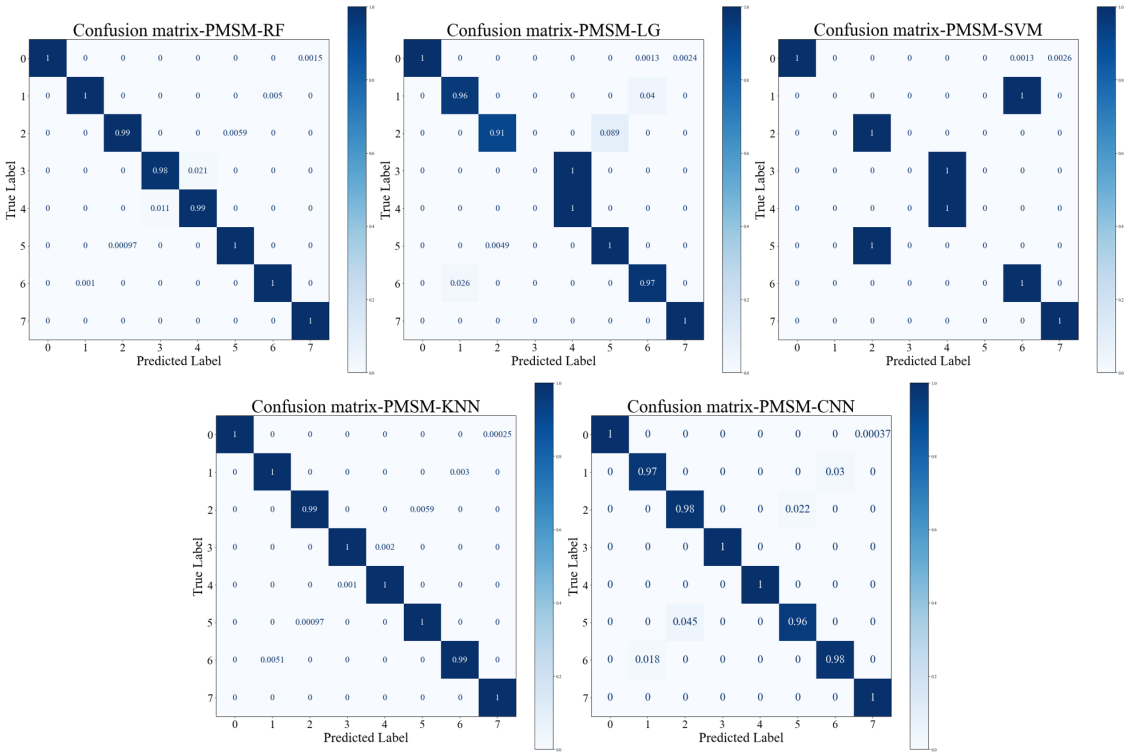


Figure 5.7: Confusion matrices for 8-class PMSM individual monitor. (0: normal condition, 1,2,3,4,5,6,7: case 1-7)

ACIM Individual Monitor Level I (3 class)

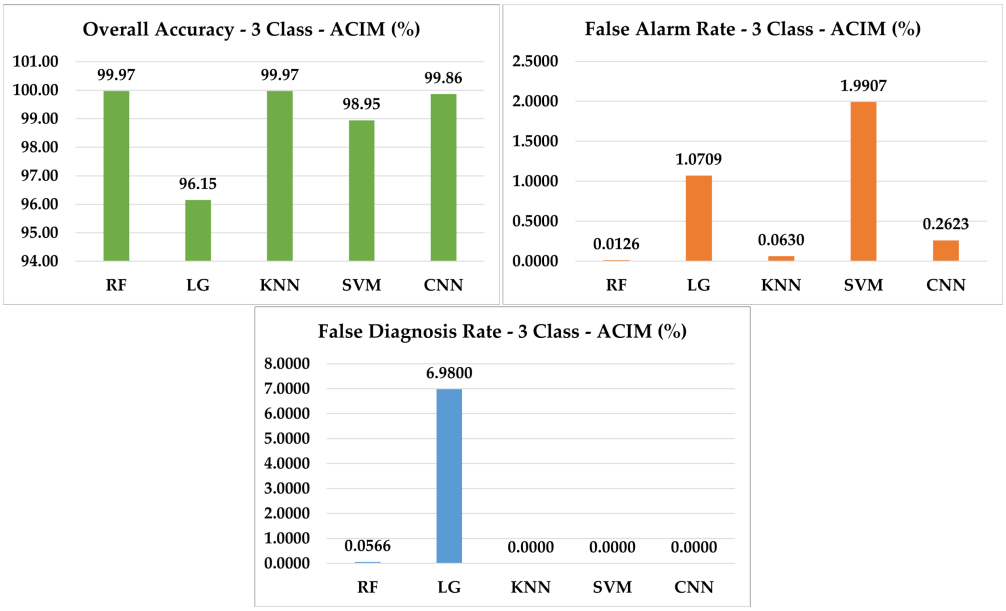


Figure 5.8: Overall accuracy, false alarm rate, and false diagnosis rate for 3-class ACIM individual monitor.

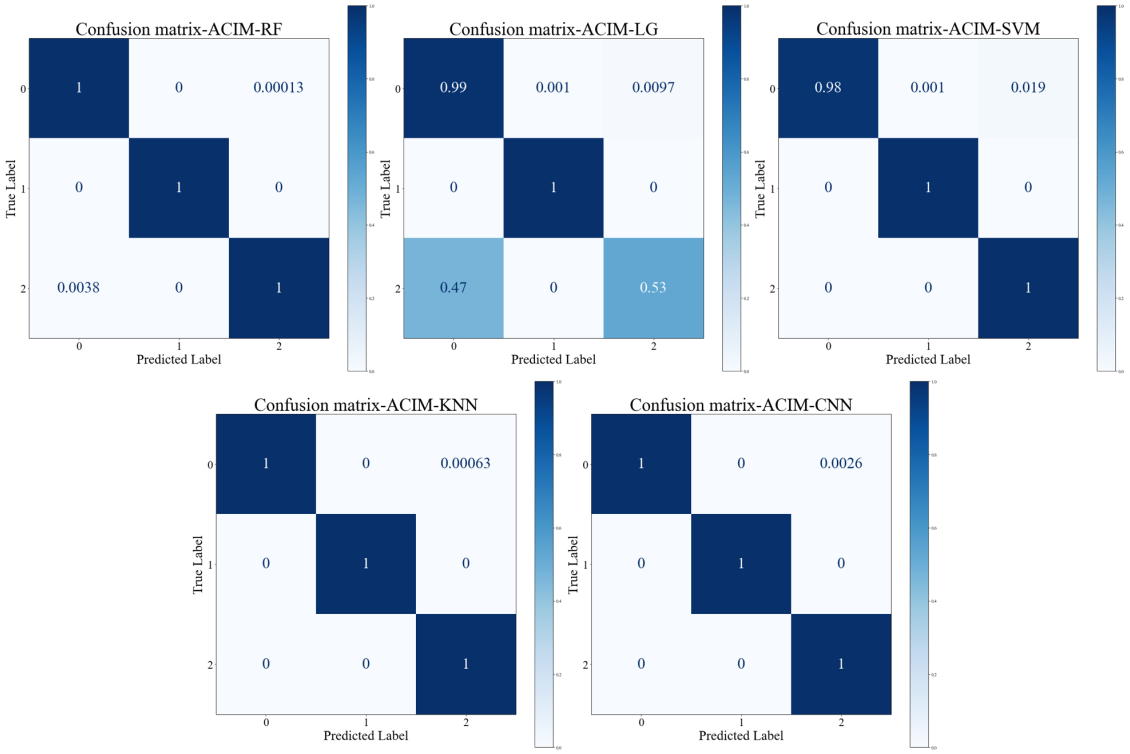


Figure 5.9: Confusion matrices for 3-class ACIM individual monitor. (0: normal condition, 1: ADC offset attacks, 2: speed reference attacks)

ACIM Individual Monitor Level II (8 class)

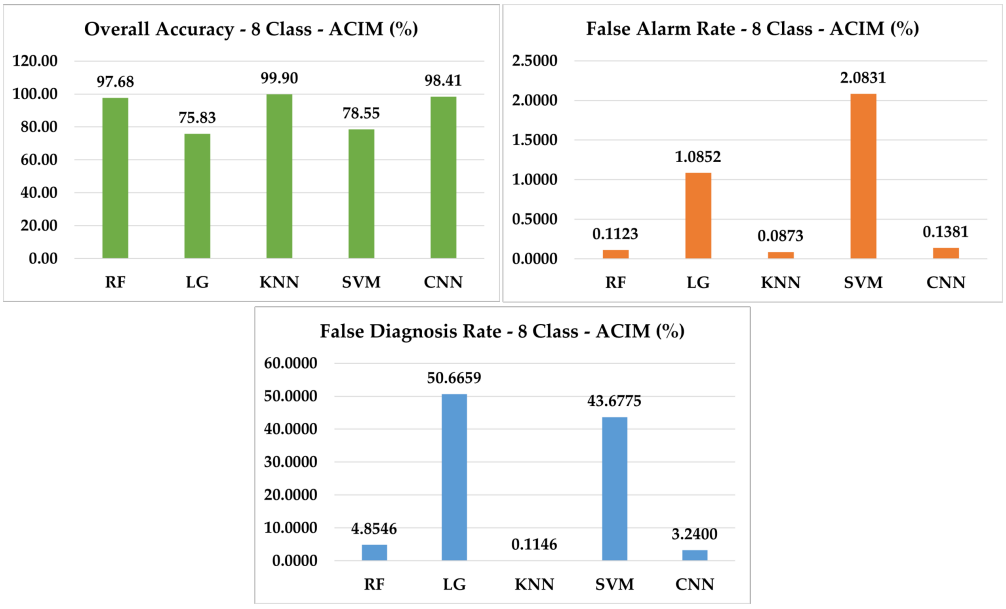


Figure 5.10: Overall accuracy, false alarm rate, and false diagnosis rate for 8-class ACIM individual monitor.

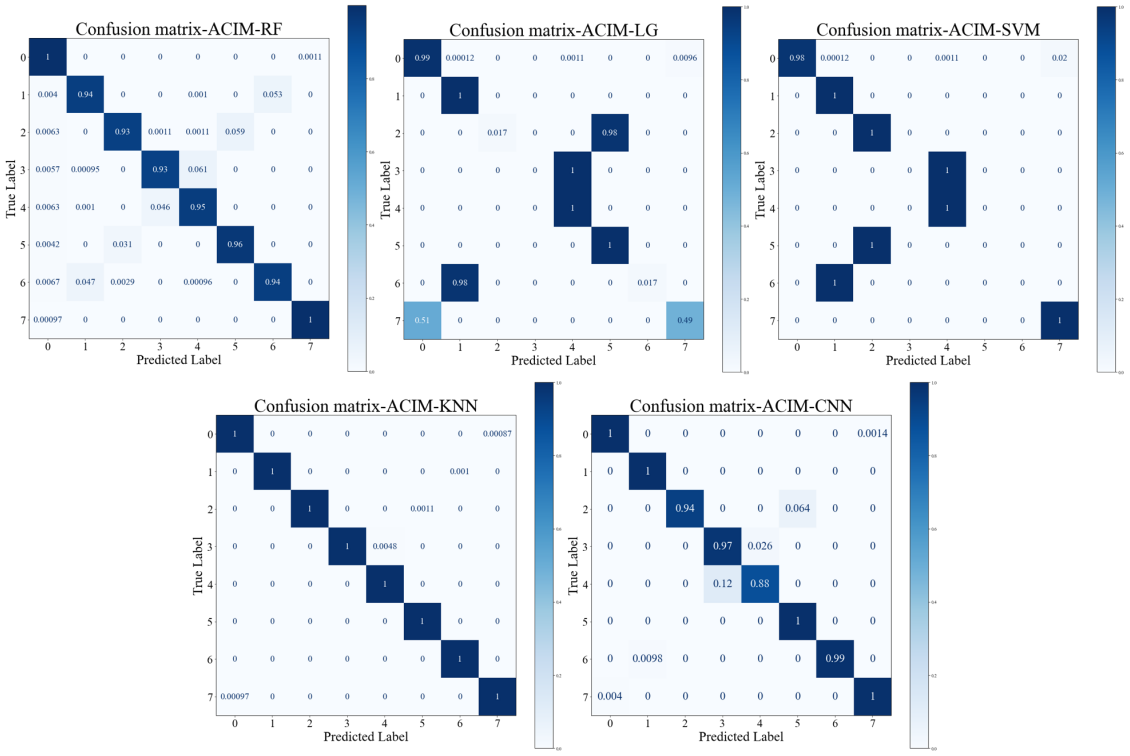


Figure 5.11: Confusion matrices for 8-class ACIM individual monitor. (0: normal condition, 1,2,3,4,5,6,7: case 8-14)

DC Bus System Monitor Level I (3 class)

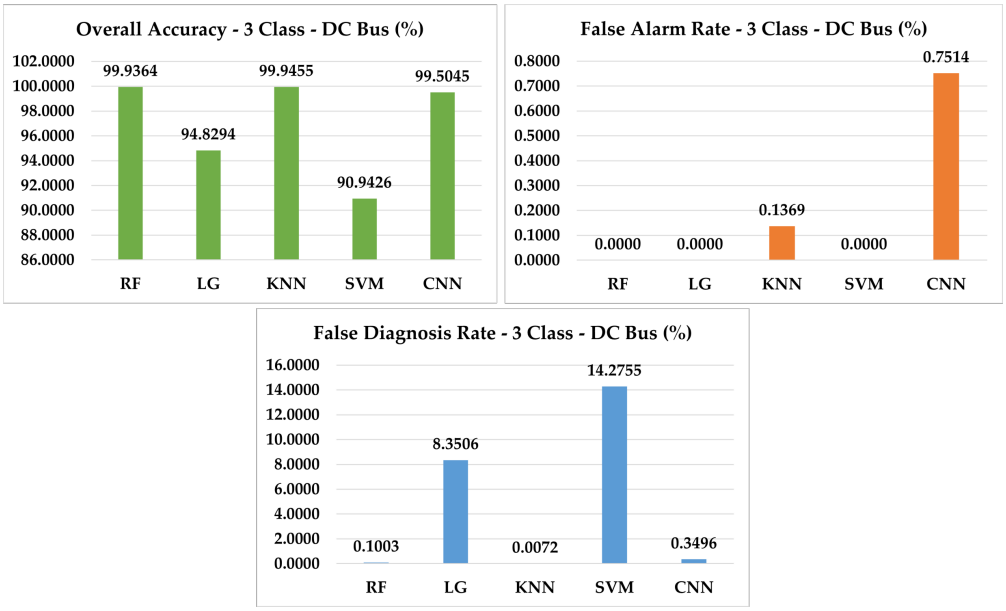


Figure 5.12: Overall accuracy, false alarm rate, and false diagnosis rate for 3-class DC bus system monitor.

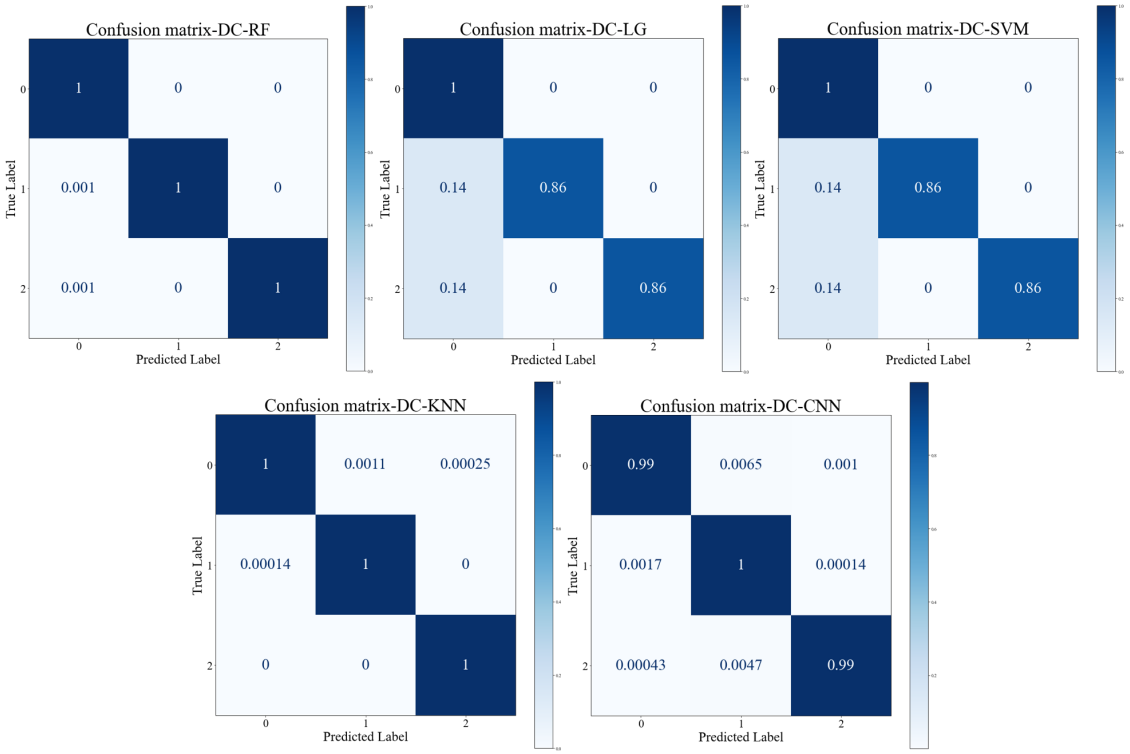


Figure 5.13: Confusion matrices for 3-class DC bus system monitor. (0: normal condition, 1: PMSM attacks, 2: ACIM attacks)

DC Bus System Monitor Level II (5 class)

Figure 5.14: Overall accuracy, false alarm rate, and false diagnosis rate for 5-class DC bus system monitor.

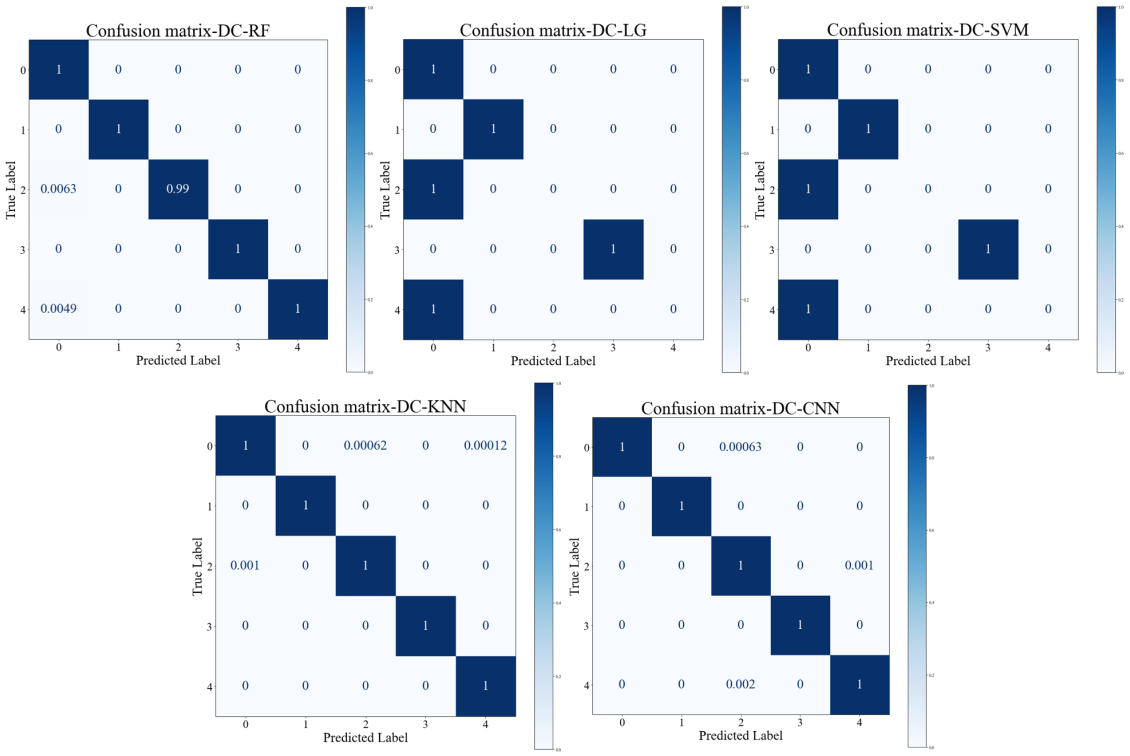


Figure 5.15: Confusion matrices for 5-class DC bus system monitor. (0: normal, 1: PMSM ADC offset attacks, 2: PMSM speed reference attacks, 3: ACIM ADC offset attacks, 4: ACIM speed reference attacks)

DC Bus System Monitor Level III (15 class)

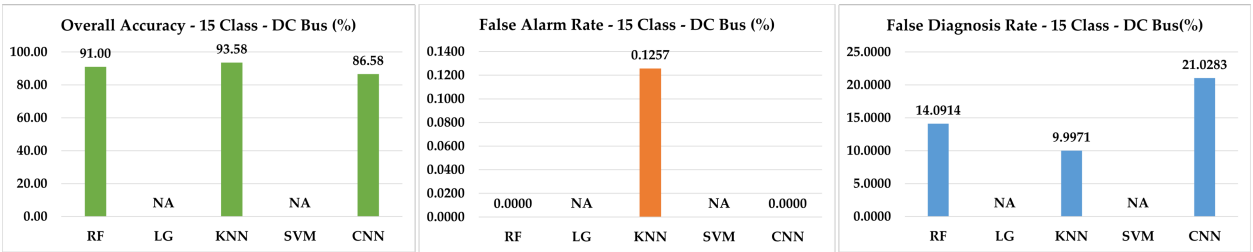


Figure 5.16: Overall accuracy, false alarm rate, and false diagnosis rate for 15-class DC bus system monitor.

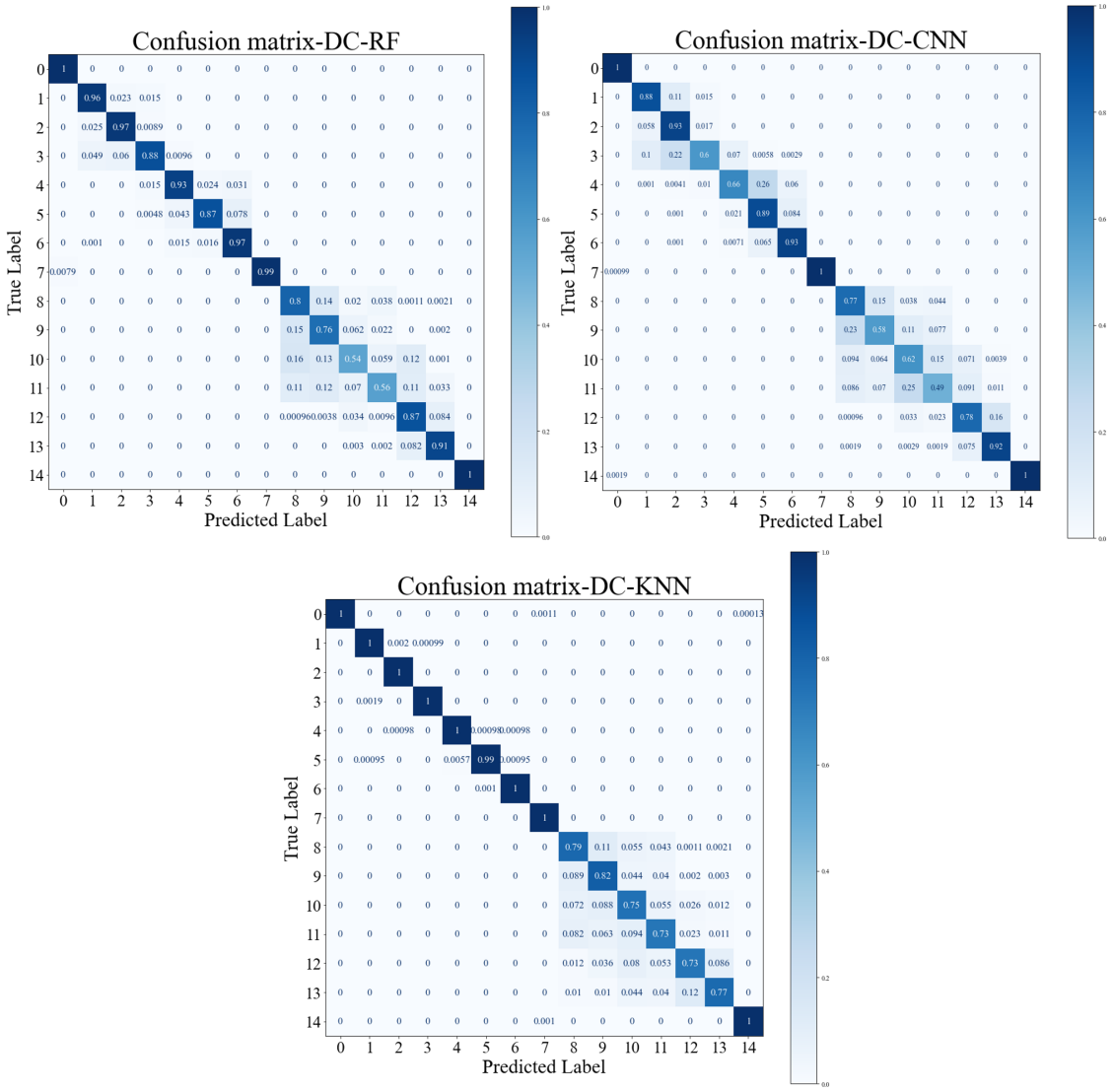


Figure 5.17: Confusion matrices for 15-class DC bus system monitor. (0: normal, 1-14: case 1-14)

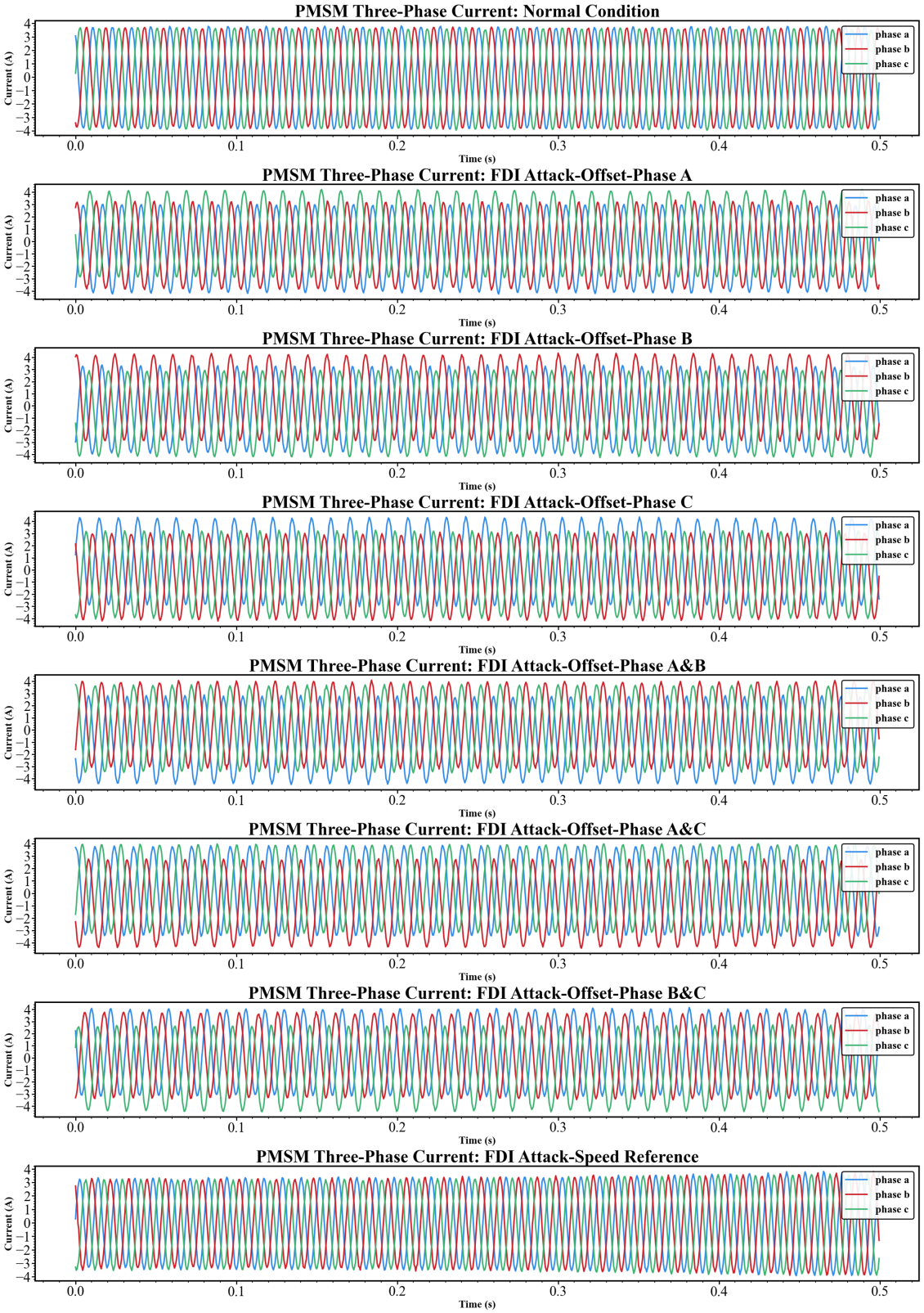


Figure 5.18: Raw waveforms of the target PMSM unit under different conditions.

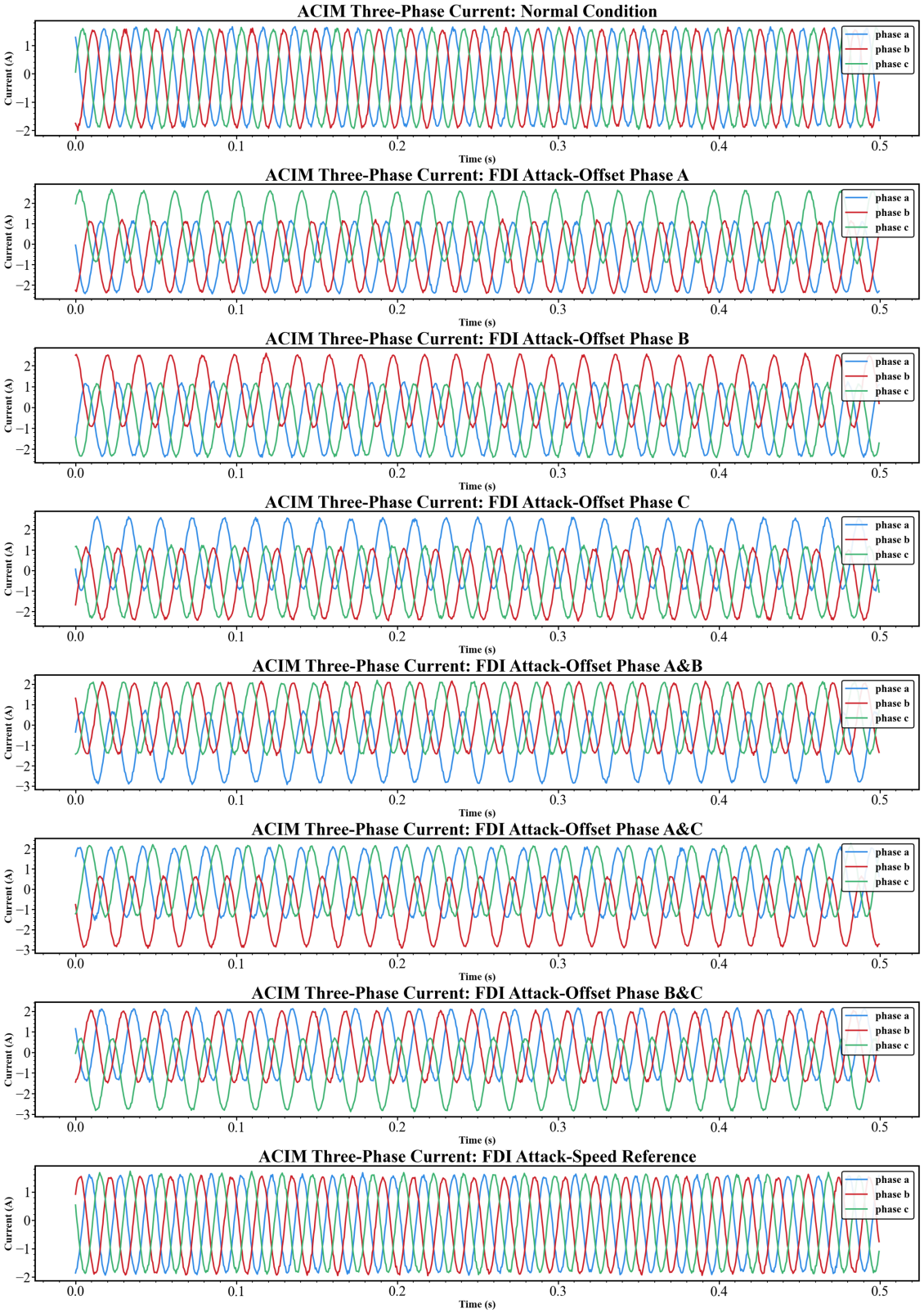


Figure 5.19: Raw waveforms of the target ACIM unit under different conditions.

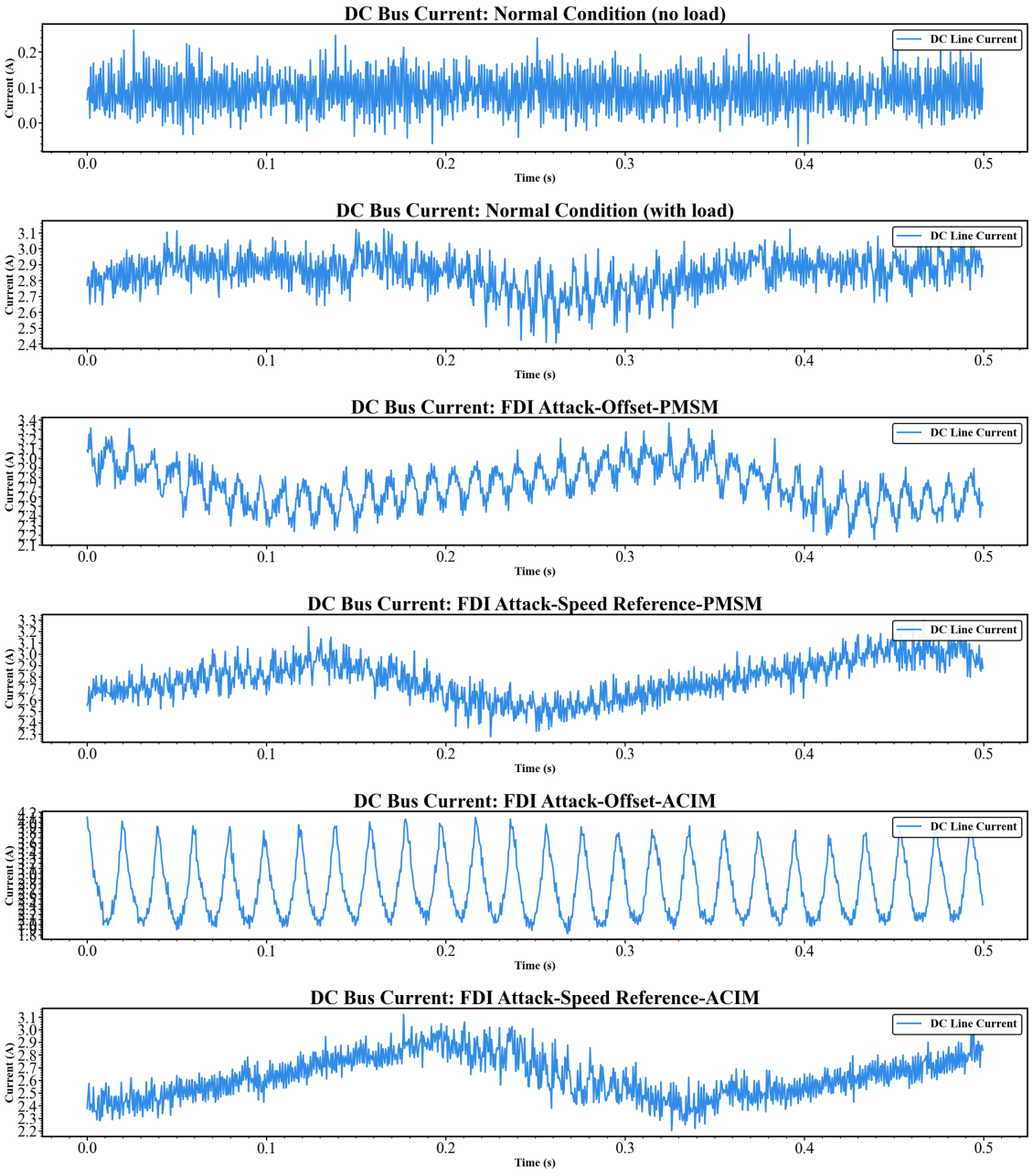


Figure 5.20: Raw waveforms of the DC bus current under different conditions.

Chapter 6

Conclusion

6.1 Conclusions and Contributions

In conclusion, this dissertation has emphasized the growing significance of cybersecurity in power systems, electric vehicles, and intelligent manufacturing systems as these domains increasingly depend on digital technologies and interconnectivity. Potential vulnerabilities within these systems encompass attacks targeting communication networks, supervisory control systems, and industrial control systems. While various strategies, including secure communication protocols, access control mechanisms, and intrusion detection systems, have been proposed to alleviate the risks associated with cyberattacks, monitoring and addressing emerging threats remains a formidable challenge as these systems continue to evolve and become more interconnected. Furthermore, this study has underscored the pivotal role that electric drive systems play in power systems, electric vehicles, and intelligent manufacturing systems. However, the cybersecurity of these systems has garnered relatively little attention. To address this gap, the present research contributes to three key areas: (1) the cyber-physical modeling of intelligent electric drive systems, (2) the development of a cyber-physical testbed for modern industrial motor drives, and (3) the implementation of advanced anomaly detection and root-cause diagnosis algorithms.

The proposed methodology for modeling intelligent electric drive systems integrates the physics-equation-based model and the control information flow model, culminating in an analytical framework that encompasses both physical faults and cyberattacks. This framework enables the mathematical formulation of system behaviors under various physical fault and cyberattack scenarios. Moreover, the framework facilitates seamless connections between existing modeling approaches from the cyber-domain, such as state machine models, and approaches from the physical domain, including state-space models. Consequently, the proposed modeling approach significantly contributes to bridging the gaps between current modeling methods employed by different communities.

The development of a cyber-physical testbed for modern industrial motor drives encompasses two aspects: a hardware-in-the-loop (HIL) real-time simulation testbed and a lab-scale real-world hardware experiment testbed. Each testbed addresses distinct facets of cyber-physical security and safety research pertaining to intelligent electric drive systems. The HIL testbed concentrates on mitigating potential risks

and costs during the research and development process concerning cyberattacks and physical faults. In conjunction with the proposed cyber-physical modeling approaches, this testbed can generate a considerable volume of high-quality data related to various attack and fault scenarios. Moreover, the HIL testbed facilitates testing and validating controller designs without the risk of causing real-world damages, such as those involving high-power equipment. On the other hand, the lab-scale hardware experiment testbed focuses on real-world data generation, system final validation, and prototype demonstration. This real-world experiment testbed yields more persuasive results based on the analysis and development derived from the HIL testbed and plays a crucial role in validating outcomes from simulation environments.

Additionally, the development of data-driven anomaly detection and root-cause diagnosis methods addresses four critical problems in existing monitoring systems for both cyberattacks and physical faults. Firstly, the adoption of easy-to-calculate time-domain features reduces the time-to-detect by more than 50% compared to traditional spectrum-based approaches. Secondly, the proposed method addresses the challenge of distinguishing between stealthy cyberattacks and physical faults in the diagnostic process by combining advanced data-driven methods and motor line current signatures, achieving a diagnostic accuracy of over 95%. The third method targets the challenge of monitoring multiple motor drives with limited sensors by strategically placing the monitoring current sensors at the point-of-common-coupling (PCC) for AC-powered systems, or at the DC bus for DC-powered systems. In this study, the proposed method successfully achieves an accuracy of more than 93% with only three (for AC-powered systems) or one (for DC-powered systems) monitoring sensors for a four-motor electric drive system (originally requiring 12 monitoring sensors). Lastly, the fourth proposed method aims to reduce dependence on large amounts of high-cost and high-risk experimental data sets. This method combines transfer learning techniques, convolutional neural networks, and motor current signatures, successfully reducing the required experimental data to 10% of the previously required size while maintaining similar detection and diagnostic accuracy.

Ultimately, a prototype condition monitoring system is developed by synthesizing all the previous analyses and advancements presented in this study, culminating in a comprehensive demonstration of a novel monitoring platform for future intelligent electric drive systems.

In summary, this dissertation has furnished a comprehensive approach to tackle the cybersecurity challenges confronted by electric drive systems in power systems, electric vehicles, and intelligent manufacturing systems. The proposed methodology offers a promising direction for future research and development efforts, aiming to enhance the cybersecurity of these vital systems.

6.2 Future Work

The world of modern electric drive systems is at a pivotal moment, with an increasing demand for intelligence, security, reliability, and resilience. As a result, in addition to the traditional research focus on advanced system design and control for higher efficiency and power density, attention is also shifting towards system security, reliability, and resilience. Advancing these aspects will require connecting knowledge from different areas, as these topics are highly interdisciplinary. For my future research, I plan to

continue conducting fundamental investigations into intelligent power electronics and electric machines. More specifically, I aim to explore the following three directions: (1) hybrid modeling approach for power-electronics-dominant complex systems, (2) physics-guided machine learning for power electronics and electric machines, and (3) decentralized power systems.

The hybrid modeling approach aims to unify the cyber and physical domains of future power electronics systems. As the process of electrification and decarbonization continues, power electronics will increasingly dominate complex systems such as power systems, electric vehicles, and industrial manufacturing systems. Consequently, integrating various aspects of future intelligent power electronics systems, including system concept, primary and secondary control development, and communication protocol design, will be essential. Since these aspects involve various dynamics, such as continuous, discrete, agent-based, and data-driven, a robust hybrid model is desirable to enable engineers to efficiently design system architectures, analyze system behaviors, and identify vulnerabilities, among other tasks.

The physics-guided machine learning approach seeks to further integrate cutting-edge data-driven methodologies into intelligent power electronics and electric machines. As most power electronics-based systems are safety-critical and mission-critical, traditional machine learning methods, which essentially function as black boxes, are challenging to implement in real-world applications. By incorporating existing physical knowledge to make data-driven algorithms more predictable, interpretable, and reliable, future power electronics systems can better leverage the advancements in machine learning.

Finally, with the widespread implementation of inverter-based distributed energy sources in modern power systems, decentralization presents a promising pathway towards enhanced security, reliability, and resilience. Although decentralization seems feasible with existing technologies such as blockchain, metaverse, and cryptocurrency, decentralizing modern power systems is no easy task. Numerous crucial factors must be considered, including system stability, fault and attack resilience, and system robustness. However, challenges also present opportunities, and I am excited to explore the potential of these futuristic research directions.

Appendix A

Publications: Peer-Reviewed Journal

1. **B. Yang** S. Wu, J. Ye, W. Song, P. Ma, J. Shi and P. Liu "Cyber-Attack Detection for Intelligent Motor Drives Based on Convolutional Neural Network and Transfer Learning," in *IEEE Transactions on Power Electronics*, 2023 (Under Review).

2. **B. Yang** and J. Ye, "A New Security Framework for Electric Machine Drives," in *IEEE Transactions on Power Electronics*, 2023 (Under Review).

3. **B. Yang**, L. Guo, F. Li, J. Ye and W. Song, "Vulnerability Assessments of Electric Drive Systems Due to Sensor Data Integrity Attacks," in *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3301-3310, May 2020.

4. **B. Yang**, J. Ye and L. Guo, "Fast Detection for Cyber Threats in Electric Vehicle Traction Motor Drives," in *IEEE Transactions on Transportation Electrification*, vol. 8, no. 1, pp. 767-777, March 2022.

5. L. Guo, **B. Yang** and J. Ye, "Predictive Energy Management for Dual-Motor BEVs Considering Temperature - Dependent Traction Inverter Loss," in *IEEE Transactions on Transportation Electrification*, vol. 8, no. 1, pp. 1501-1515, March 2022.

6. L. Guo, **B. Yang**, J. Ye, J. M. Velni and W. Song, "Attack-Resilient Lateral Stability Control for Four-Wheel-Driven EVs Considering Changed Driver Behavior Under Cyber Threats," in *IEEE Transactions on Transportation Electrification*, vol. 8, no. 1, pp. 1362-1375, March 2022.

7. L. Guo, **B. Yang**, J. Ye, H. Chen, F. Li, W. Song, L. Du, and L. Guan, "Systematic Assessment of Cyber-Physical Security of Energy Management System for Connected and Automated Electric Vehicles," in *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3335-3347, May 2021.

8. L. Guo, J. Ye and **B. Yang**, "Cyberattack Detection for Electric Vehicles Using Physics-Guided Machine Learning," in *IEEE Transactions on Transportation Electrification*, vol. 7, no. 3, pp. 2010-2022, Sept. 2021.

9. J. Ye, L. Guo, **B. Yang**, F. Li, L. Du, L. Guan, and W. Song, "Cyber-Physical Security of Powertrain Systems in Modern Electric Vehicles: Vulnerabilities, Challenges, and Future Visions," in *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 4, pp. 4639-4657, Aug. 2021.

10. F. Li, R. Xie, **B. Yang**, L. Guo, P. Ma, J. Shi, J. Ye, and W. Song, "Detection and Identification of Cyber and Physical Attacks on Distribution Power Grids with PVs: An Online High-Dimensional Data-driven Approach," in *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 10, no. 1, pp. 1282-1291, Feb. 2022.

Appendix B

Publications: Peer-Reviewed Conference

1. S. J. Coshatt, **B. Yang**, J. Ye, W. Song, F. Zahiri and J. Hill, "Fault and Attack Detection and Diagnosis by Analysis of Electrical Waveforms of Power Networks," in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1-9.

2. **B. Yang** and J. Ye, "Data-Driven Detection of Physical Faults and Cyber Attacks in Dual-Motor EV Powertrains," in *2022 IEEE Transportation Electrification Conference & Expo (ITEC)*, 2022, pp. 991-996.

3. **B. Yang**, J. Ye, S. Coshatt, W. Song and F. Zahiri, "Data-Driven Approach for Detection of Physical Faults and Cyber Attacks in Manufacturing Motor Drives," in *2022 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2022, pp. 1-6.

4. S. Coshatt, Q. Li, **B. Yang**, S. Wu, D. Shrivastava, J. Ye, W. Song and F. Zahiri, "Design of Cyber-Physical Security Testbed for Multi-Stage Manufacturing System," in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, 2022, pp. 1978-1983.

5. **B. Yang**, L. Guo and J. Ye, "Physics-Based Attack Detection for Traction Motor Drives in Electric Vehicles Using Random Forest," in *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2021, pp. 849-854.

6. **B. Yang**, L. Guo, J. Ye and J. M. Velni, "Energy Management Strategy for Dual-Motor-Based Electric Vehicle Powertrain Using Nonlinear Model Predictive Control," in *2021 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2021, pp. 206-211.

7. L. Guo, **B. Yang** and J. Ye, "Detection and Diagnosis of Long-Term Cyber-Attacks for Predictive Energy Management System in HEVs," in *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2021, pp. 842-848.

8. L. Guo, **B. Yang**, J. Ye and J. M. Velni, "Attack-Resilient Lateral Stability Control for Autonomous In-Wheel-Motor-Driven Electric Vehicles," in *2021 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2021, pp. 200-205.

9. **B. Yang**, L. Guo and J. Ye, "Real-time Simulation of Electric Vehicle Powertrain: Hardware-in-the-Loop (HIL) Testbed for Cyber-Physical Security," in *2020 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2020, pp. 63-68.

10. L. Guo, **B. Yang** and J. Ye, "Enhanced Cyber-physical Security of Steering Stability Control System for Four-Wheel Independent Drive Electric Vehicles," in *2020 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2020, pp. 1240-1245.

11. **B. Yang**, L. Guo, F. Li, J. Ye and W. Song, "Impact Analysis of Data Integrity Attacks on Power Electronics and Electric Drives," in *2019 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2019, pp. 1-6.

12. **B. Yang**, F. Li, J. Ye and W. Song, "Condition Monitoring and Fault Diagnosis of Generators in Power Networks," in *2019 IEEE Power & Energy Society General Meeting (PESGM)*, 2019, pp. 1-5.

Bibliography

Anwar, A., Mahmood, A. N., & Shah, Z. (2015). A data-driven approach to distinguish cyber-attacks from physical faults in a smart grid. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 1811–1814.

Araújo, J. F. B., Pedrosa, H. M., Rodrigues, M. C. B. P., & Barbosa, P. G. (2016). Real-time hardware-in-the-loop simulation of components of an electric vehicle powertrain: Modeling and implementation. *2016 12th IEEE International Conference on Industry Applications*, 1–7.

Austin, T. H., & Flanagan, C. (2009). Efficient purely-dynamic information flow analysis. *Proceedings of the ACM SIGPLAN Fourth Workshop on Programming Languages and Analysis for Security*, 113–124.

Awadallah, M., Tawadros, P., Walker, P., & Zhang, N. (2018). Hardware-in-the-loop simulation for the design and testing of motor in advanced powertrain applications. *2018 IEEE 27th International Symposium on Industrial Electronics*, 817–824.

Bahill, A. T., & Szidarovszky, F. (2009). Comparison of dynamic system modeling methods. *Systems Engineering*, *12*(3), 183–200.

Baskaya, E., Bronz, M., & Delahaye, D. (2017). Fault detection & diagnosis for small uavs via machine learning. *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, 1–6.

Beg, O. A., Nguyen, L. V., Johnson, T. T., & Davoudi, A. (2021). Cyber-physical anomaly detection in microgrids using time-frequency logic formalism. *IEEE Access*, *9*, 20012–20021. https://doi.org/10.1109/ACCESS.2021.3055229

Beheshtaein, S., Cuzner, R., Savaghebi, M., & Guerrero, J. M. (2019). Review on microgrids protection. *IET Generation, Transmission & Distribution*, *13*(6), 743–759.

Berthier, R., Sanders, W. H., & Khurana, H. (2010). Intrusion detection for advanced metering infrastructures: Requirements and architectural directions. *2010 First IEEE International Conference on Smart Grid Communications*, 350–355.

Bezemskij, A., Loukas, G., Gan, D., & Anthony, R. J. (2017). Detecting cyber-physical threats in an autonomous robotic vehicle using bayesian networks. *2017 IEEE International Conference on Internet of Things and IEEE Green Computing and Communications and IEEE Cyber, Physical and Social Computing and IEEE Smart Data*, 98–103.

Bhamare, D., Zolanvari, M., Erbad, A., Jain, R., Khan, K., & Meskin, N. (2020). Cybersecurity for industrial control systems: A survey. *Computers & Security*, *89*, 101677.

Bharati, S., Podder, P., Mondal, M., Robel, M., & Alam, R. (2020). Threats and countermeasures of cyber security in direct and remote vehicle communication systems. *arXiv preprint arXiv:2006.08723*.

Bo, P., Granato, A., Mancuso, M. E., Ciccotelli, C., & Querzoni, L. (2019). Fada-cps—faults and attacks discrimination in cyber physical systems. In *Policy-based autonomic data governance* (pp. 91–112). Springer.

Broggi, A., Cerri, P., Felisa, M., Laghi, M. C., Mazzei, L., & Porta, P. P. (2012). The vislab intercontinental autonomous challenge: An extensive test for a platoon of intelligent vehicles. *International Journal of Vehicle Autonomous Systems*, *10*(3), 147–164.

Canaan, B., Colicchio, B., & Ould Abdeslam, D. (2020). Microgrid cyber-security: Review and challenges toward resilience. *Applied Sciences*, *10*(16), 5649.

Cárdenas, A. A., Amin, S., Lin, Z.-S., Huang, Y.-L., Huang, C.-Y., & Sastry, S. (2011). Attacks against process control systems: Risk assessment, detection, and response. *Proceedings of the 6th ACM symposium on information, computer and communications security*, 355–366.

Cárdenas, A. A., Amin, S., & Sastry, S. (2008). Research challenges for the security of control systems. *HotSec*, *5*, 15.

Charette, R. (2009). This car runs on code.

Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S., Koscher, K., Czeskis, A., Roesner, F., Kohno, T., et al. (2011). Comprehensive experimental analyses of automotive attack surfaces. *USENIX security symposium*, *4*(447-462), 2021.

Chen, S., Ge, H., Li, J., & Pecht, M. (2019). Progressive improved convolutional neural network for avionics fault diagnosis. *IEEE Access*, *7*, 177362–177375.

Cherry, S., & Langner, R. (2010). How stuxnet is rewriting the cyberterrorism playbook. *Computerworld*.

Choi, S., Haque, M. S., Arafat, A., & Toliyat, H. A. (2017). Detection and estimation of extremely small fault signature by utilizing multiple current sensor signals in electric machines. *IEEE Transactions on Industry Applications*, *53*(3), 2805–2816.

Cui, S., Han, Z., Kar, S., Kim, T. T., Poor, H. V., & Tajer, A. (2012). Coordinated data-injection attack and detection in the smart grid: A detailed look at enriching detection solutions. *IEEE Signal Processing Magazine*, *29*(5), 106–115.

Culbert, I., & Rhodes, W. (2005). Using current signature analysis technology to reliably detect cage winding defects in squirrel cage induction motors. *Record of Conference Papers Industry Applications Society 52nd Annual Petroleum and Chemical Industry Conference*, 95–101.

Cyber attack in a two-area power system: Impact identification using reachability. (2010). *Proceedings of the 2010 American Control Conference*, 962–967. https://doi.org/10.1109/ACC.2010.5530460

Dan, G., & Sandberg, H. (2010). Stealth attacks and protection schemes for state estimators in power systems. *2010 first IEEE international conference on smart grid communications*, 214–219.

Dehghani, M., Niknam, T., Ghiasi, M., Bayati, N., & Savaghebi, M. (2021). Cyber-attack detection in dc microgrids based on deep machine learning and wavelet singular values approach. *Electronics*, *10*(16), 1914.

Dexter, A. (1995). Fuzzy model based fault diagnosis. *IEE Proceedings-Control Theory and Applications*, *142*(6), 545–550.

Drias, Z., Serhrouchni, A., & Vogel, O. (2015). Taxonomy of attacks on industrial control protocols. *2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS)*, 1–6. https://doi.org/10.1109/NOTERE.2015.7293513

Elçi, A., Koné, M. T., & Orgun, M. A. (2011). *Semantic agent systems: Foundations and applications* (Vol. 344). Springer Science & Business Media.

Eldin, N., et al. (1994). Explicit modelling of the stator winding bar water cooling for model-based fault diagnosis of turbogenerators with experimental verification. *1994 Proceedings of IEEE International Conference on Control and Applications*, 1403–1408.

Final report on the august 14, 2003 blackout in the united states and canada: Causes and recommendations (tech. rep.). (2004). U.S.-Canada Power System Outage Task Force.

Giraldo, J., Urbina, D., Cardenas, A., Valente, J., Faisal, M., Ruths, J., Tippenhauer, N. O., Sandberg, H., & Candell, R. (2018). A survey of physics-based attack detection in cyber-physical systems. *ACM Computing Surveys (CSUR)*, *51*(4), 1–36.

Guo, L., Yang, B., & Ye, J. (2020). Enhanced cyber-physical security of steering stability control system for four-wheel independent drive electric vehicles. *2020 IEEE Transportation Electrification Conference Expo (ITEC)*, 1240–1245.

Guo, L., Yang, B., Ye, J., Chen, H., Li, F., Song, W., Du, L., & Guan, L. (2020). Systematic assessment of cyber-physical security of energy management system for connected and automated electric vehicles. *IEEE Transactions on Industrial Informatics*, 1–1.

Guo, L., & Ye, J. (2020). Cyber-physical security of electric vehicles with four motor drives. *IEEE Transactions on Power Electronics*, 1–1.

Guo, L., Yang, B., & Ye, J. (2021). Detection and diagnosis of long-term cyber-attacks for predictive energy management system in hevs. *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 842–848. https://doi.org/10.1109/APEC42165.2021.9487077

Guo, L., Yang, B., Ye, J., & Velni, J. M. (2021). Attack-resilient lateral stability control for autonomous in-wheel-motor-driven electric vehicles. *2021 IEEE Transportation Electrification Conference Expo (ITEC)*, 200–205. https://doi.org/10.1109/ITEC51675.2021.9490133

Guo, L., Yang, B., Ye, J., Velni, J. M., & Song, W. (2021). Attack-resilient lateral stability control for four-wheel-driven evs considering changed driver behavior under cyber threats. *IEEE Transactions on Transportation Electrification*, 1–1. https://doi.org/10.1109/TTE.2021.3102134

Guo, L., Yang, B., Ye, J., Velni, J. M., & Song, W. (2022). Attack-resilient lateral stability control for four-wheel-driven evs considering changed driver behavior under cyber threats. *IEEE Transactions on Transportation Electrification*, *8*(1), 1362–1375. https://doi.org/10.1109/TTE.2021.3102134

Guo, L., Ye, J., & Yang, B. (2021). Cyberattack detection for electric vehicles using physics-guided machine learning. *IEEE Transactions on Transportation Electrification*, *7*(3), 2010–2022. https://doi.org/10.1109/TTE.2020.3044524

Gupta, K., Sahoo, S., Mohanty, R., Panigrahi, B. K., & Blaabjerg, F. (2022). Distinguishing between cyber attacks and faults in power electronic systems - a non-invasive approach. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 1–1. https://doi.org/10.1109/JESTPE.2022.3221867

Habibi, M. R., Baghaee, H. R., Dragičević, T., & Blaabjerg, F. (2021). Detection of false data injection cyber-attacks in dc microgrids based on recurrent neural networks. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *9*(5), 5294–5310. https://doi.org/10.1109/JESTPE.2020.2968243

Han, S., Xie, M., Chen, H.-H., & Ling, Y. (2014). Intrusion detection in cyber-physical systems: Techniques and challenges. *IEEE systems journal*, *8*(4), 1052–1062.

Hongyu, W., Zhiqiang, Y., Yong, F., Yunqian, Q., & Zhiming, L. (2015). Development of pure electric vehicle powertrain controller based on hardware in the loop platform. *2015 6th IEEE International Conference on Software Engineering and Service Science*, 498–502. https://doi.org/10.1109/ICSESS.2015.7339106

Hounsinou, S., Stidd, M., Ezeobi, U., Olufowobi, H., Nasri, M., & Bloom, G. (2021). Vulnerability of controller area network to schedule-based attacks. *2021 IEEE Real-Time Systems Symposium (RTSS)*, 495–507. https://doi.org/10.1109/RTSS52674.2021.00051

Huang, Y.-L., Cárdenas, A. A., Amin, S., Lin, Z.-S., Tsai, H.-Y., & Sastry, S. (2009). Understanding the physical and economic consequences of attacks on control systems. *International Journal of Critical Infrastructure Protection*, *2*(3), 73–83.

Humayed, A., Lin, J., Li, F., & Luo, B. (2017). Cyber-physical systems security—a survey. *IEEE Internet of Things Journal*, *4*(6), 1802–1831. https://doi.org/10.1109/JIOT.2017.2703172

Ilascu, I. (2019). Hvacking: Remotely exploiting bugs in building control systems.

Imai, S., Chen, S., Zhu, W., & Varela, C. A. (2019). Dynamic data-driven learning for self-healing avionics. *Cluster Computing*, *22*(1), 2187–2210.

Jafarnejad, S., Codeca, L., Bronzi, W., Frank, R., & Engel, T. (2015). A car hacking experiment: When connectivity meets vulnerability. *2015 IEEE Globecom Workshops (GC Wkshps)*, 1–6. https://doi.org/10.1109/GLOCOMW.2015.7413993

Kar, C., & Mohanty, A. (2006). Monitoring gear vibrations through motor current signature analysis and wavelet transform. *Mechanical systems and signal processing*, *20*(1), 158–187.

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., & Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, *3*(6), 422–440.

Karnouskos, S. (2011). Stuxnet worm impact on industrial cyber-physical system security. *IECON 2011 - 37th Annual Conference of the IEEE Industrial Electronics Society*, 4490–4494. https://doi.org/10.1109/IECON.2011.6120048

Khan, A. A., Beg, O. A., Alamaniotis, M., & Ahmed, S. (2021). Intelligent anomaly identification in cyber-physical inverter-based systems. *Electric Power Systems Research*, *193*, 107024.

Kim, K., & Parlos, A. G. (2002). Induction motor fault diagnosis based on neuropredictors and wavelet signal processing. *IEEE/ASME Transactions on mechatronics*, *7*(2), 201–219.

Kiss, I., Genge, B., & Haller, P. (2015). A clustering-based approach to detect cyber attacks in process control systems. *2015 IEEE 13th international conference on industrial informatics (INDIN)*, 142–148.

Kliman, G., & Stein, J. (1992). Methods of motor current signature analysis. *Electric Machines and power systems*, *20*(5), 463–474.

Koppel, T. (2016). *Lights out: A cyberattack, a nation unprepared, surviving the aftermath*. Crown.

Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., et al. (2010). Experimental security analysis of a modern automobile. *2010 IEEE Symposium on Security and Privacy*, 447–462.

Kwon, C., Liu, W., & Hwang, I. (2013). Security analysis for cyber-physical systems against stealthy deception attacks. *2013 American control conference*, 3344–3349.

Lab, T. K. S. (2019). *Experimental security research of tesla autopilot* (tech. rep.). Keen Security Lab.

Langner, R. (2011). Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, *9*(3), 49–51.

Leitao, P., Karnouskos, S., Ribeiro, L., Lee, J., Strasser, T., & Colombo, A. W. (2016). Smart agents in industrial cyber-physical systems. *Proceedings of the IEEE*, *104*(5), 1086–1101. https://doi.org/10.1109/JPROC.2016.2521931

Li, F., Li, Q., Zhang, J., Kou, J., Ye, J., Song, W., & Mantooth, H. A. (2021). Detection and diagnosis of data integrity attacks in solar farms based on multilayer long short-term memory network. *IEEE Transactions on Power Electronics*, *36*(3), 2495–2498. https://doi.org/10.1109/TPEL.2020.3017935

Li, F., Shi, Y., Shinde, A., Ye, J., & Song, W.-Z. (2019). Enhanced cyber-physical security in internet of things through energy auditing. *IEEE Internet of Things Journal*, *6*(3), 5224–5231.

Li, F., Shinde, A., Shi, Y., Ye, J., Li, X.-Y., & Song, W.-Z. (2019). System statistics learning-based iot security: Feasibility and suitability. *IEEE Internet of Things Journal*, *6*(4), 6396–6403.

Li, Q., Li, F., Zhang, J., Ye, J., Song, W., & Mantooth, A. (2020). Data-driven cyberattack detection for photovoltaic (pv) systems through analyzing micro-pmu data. *2020 IEEE Energy Conversion Congress and Exposition (ECCE)*, 431–436. https://doi.org/10.1109/ECCE44975.2020.9236274

Li, Q., Zhang, J., Ye, J., & Song, W. (2022). Data-driven cyber-attack detection for photovoltaic systems: A transfer learning approach. *2022 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 1926–1930. https://doi.org/10.1109/APEC43599.2022.9773401

Li, X., Zhang, W., Ding, Q., & Li, X. (2020). Diagnosing rotating machines with weakly supervised data using deep transfer learning. *IEEE Transactions on Industrial Informatics*, *16*(3), 1688–1697. https://doi.org/10.1109/TII.2019.2927590

Liao, P., Yan, J., Sellier, J. M., & Zhang, Y. (2022). Divergence-based transferability analysis for self-adaptive smart grid intrusion detection with transfer learning. *IEEE Access*, *10*, 68807–68818. https://doi.org/10.1109/ACCESS.2022.3186328

Liao, Y., Deschamps, F., Loures, E. d. F. R., & Ramos, L. F. P. (2017). Past, present and future of industry 4.0-a systematic literature review and research agenda proposal. *International journal of production research*, *55*(12), 3609–3629.

Loukas, G., Karapistoli, E., Panaousis, E., Sarigiannidis, P., Bezemskij, A., & Vuong, T. (2019). A taxonomy and survey of cyber-physical intrusion detection approaches for vehicles. *Ad Hoc Networks*, *84*, 124–147.

Loukas, G., Yoon, Y., Sakellari, G., Vuong, T., & Heartfield, R. (2017). Computation offloading of a vehicle's continuous intrusion detection workload for energy efficiency and performance. *Simulation Modelling Practice and Theory*, *73*, 83–94.

Martini, S., Di Baccio, D., Romero, F. A., Jiménez, A. V., Pallottino, L., Dini, G., & Ollero, A. (2015). Distributed motion misbehavior detection in teams of heterogeneous aerial robots. *Robotics and Autonomous Systems*, *74*, 30–39.

Maschler, B., & Weyrich, M. (2021). Deep transfer learning for industrial automation: A review and discussion of new techniques for data-driven machine learning. *IEEE Industrial Electronics Magazine*, *15*(2), 65–75. https://doi.org/10.1109/MIE.2020.3034884

McMillan, R. (2010). Siemens: Stuxnet worm hit industrial systems. *Computerworld*, *14*.

Miller, C., & Valasek, C. (2015). Remote exploitation of an unaltered passenger vehicle. *Black Hat USA*, *2015*(S 91), 1–91.

Ming, J., Wu, D., Xiao, G., Wang, J., & Liu, P. (2015). {Taintpipe}: Pipelined symbolic taint analysis. *24th USENIX Security Symposium (USENIX Security 15)*, 65–80.

Mitchell, R., & Chen, R. (2013). Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *44*(5), 593–604.

Pasqualetti, F., Dorfler, F., & Bullo, F. (2015). Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems. *IEEE Control Systems Magazine*, *35*(1), 110–127.

Phillips, L. R., Tejani, B., Margulies, J., Hills, J. L., Richardson, B. T., Baca, M. J., & Weiland, L. (2005). *Analysis of operations and cyber security policies for a system of cooperating flexible alternating current transmission system (facts) devices.* (tech. rep.). Sandia National Laboratories.

Rai, R., & Sahu, C. K. (2020). Driven by data or derived through physics? a review of hybrid physics guided machine learning techniques with cyber-physical system (cps) focus. *IEEE Access*, *8*, 71050–71073.

Romeral, L., Urresty, J. C., Riba Ruiz, J.-R., & Garcia Espinosa, A. (2011). Modeling of surface-mounted permanent magnet synchronous motors with stator winding interturn faults. *IEEE Transactions on Industrial Electronics*, *58*(5), 1576–1585. https://doi.org/10.1109/TIE.2010.2062480

Sabelfeld, A., & Myers, A. C. (2003). Language-based information-flow security. *IEEE Journal on selected areas in communications*, *21*(1), 5–19.

Sahoo, S., Mishra, S., Peng, J. C.-H., & Dragičević, T. (2019). A stealth cyber-attack detection strategy for dc microgrids. *IEEE Transactions on Power Electronics*, *34*(8), 8162–8174. https://doi.org/10.1109/TPEL.2018.2879886

Sahoo, S., Peng, J. C.-H., Devakumar, A., Mishra, S., & Dragičević, T. (2020). On detection of false data in cooperative dc microgrids—a discordant element approach. *IEEE Transactions on Industrial Electronics*, *67*(8), 6562–6571. https://doi.org/10.1109/TIE.2019.2938497

Skorobogatov, S. P., & Anderson, R. J. (2002). Optical fault induction attacks. *International workshop on cryptographic hardware and embedded systems*, 2–12.

Sridhar, S., & Manimaran, G. (2010). Data integrity attacks and their impacts on scada control system. *IEEE PES General Meeting*, 1–6. https://doi.org/10.1109/PES.2010.5590115

Subasi, A., Al-Marwani, K., Alghamdi, R., Kwairanga, A., Qaisar, S. M., Al-Nory, M., & Rambo, K. A. (2018). Intrusion detection in smart grid using data mining techniques. *2018 21st Saudi Computer Society National Computer Conference (NCC)*, 1–6. https://doi.org/10.1109/NCG.2018.8593124

Tabbache, B., Aboub, Y., Marouani, K., Kheloui, A., & Benbouzid, M. E. H. (2012). A simple and effective hardware-in-the-loop simulation platform for urban electric vehicles. *2012 First International Conference on Renewable Energies and Vehicular Technology*, 251–255. https://doi.org/10.1109/REVET.2012.6195279

Teixeira, A., Shames, I., Sandberg, H., & Johansson, K. H. (2015). A secure control framework for resource-limited adversaries. *Automatica*, *51*, 135–148.

Ten, C., Manimaran, G., & Liu, C. (2010). Cybersecurity for critical infrastructures: Attack and defense modeling. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, *40*(4), 853–865. https://doi.org/10.1109/TSMCA.2010.2048028

Tengler, S. (2020). Top 25 auto cybersecurity hacks: Too many glass houses to be throwing stones. *Forbes Business*.

Thomson, W. T., & Fenger, M. (2001). Current signature analysis to detect induction motor faults. *IEEE Industry Applications Magazine*, *7*(4), 26–34.

Toliyat, H. A., Nandi, S., Choi, S., & Meshgin-Kelk, H. (2012). *Electric machines: Modeling, condition monitoring, and fault diagnosis*. CRC press.

Tripp, O., Pistoia, M., Fink, S. J., Sridharan, M., & Weisman, O. (2009). Taj: Effective taint analysis of web applications. *ACM Sigplan Notices*, *44*(6), 87–97.

Valenzuela, J., Wang, J., & Bissinger, N. (2013). Real-time intrusion detection in power system operations. *IEEE Transactions on Power Systems*, *28*(2), 1052–1062. https://doi.org/10.1109/TPWRS.2012.2224144

Vuković, O., & Dán, G. (2013). Detection and localization of targeted attacks on fully distributed power system state estimation. *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 390–395.

Wang, L., & Wang, G. (2016). Big data in cyber-physical systems, digital manufacturing and industry 4.0. *International Journal of Engineering and Manufacturing (IJEM)*, *6*(4), 1–8.

Wen, L., Gao, L., & Li, X. (2019). A new deep transfer learning based on sparse auto-encoder for fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *49*(1), 136–144. https://doi.org/10.1109/TSMC.2017.2754287

Wu, D., Arkhipov, D. I., Kim, M., Talcott, C. L., Regan, A. C., McCann, J. A., & Venkatasubramanian, N. (2016). Addsen: Adaptive data processing and dissemination for drone swarms in urban sensing. *IEEE transactions on computers*, *66*(2), 183–198.

Wu, Y., Xiao, G., & Wang, M. (2020). Cascading failure analysis method of avionics based on operational process state. *IEEE Access*, *8*, 148425–148444.

Xie, L., Mo, Y., & Sinopoli, B. (2011). Integrity data attacks in power market operations. *IEEE Transactions on Smart Grid*, *2*(4), 659–666. https://doi.org/10.1109/TSG.2011.2161892

Yan, Y., Qian, Y., Sharif, H., & Tipper, D. (2012). A survey on cyber security for smart grid communications. *IEEE Communications Surveys & Tutorials*, *14*(4), 998–1010. https://doi.org/10.1109/SURV.2012.010912.00035

Yang, B., Guo, L., & Ye, J. (2020). Real-time simulation of electric vehicle powertrain: Hardware-in-the-loop (hil) testbed for cyber-physical security. *2020 IEEE Transportation Electrification Conference Expo (ITEC)*, 63–68.

Yang, B., Guo, L., Li, F., Ye, J., & Song, W. Impact analysis of data integrity attacks on power electronics and electric drives. In: IEEE Transportation Electrification Conference, Expo (ITEC), Detroit, MI, 2019, June 5.

Yang, B., Guo, L., Li, F., Ye, J., & Song, W. (2019b). Vulnerability assessments of electric drive systems due to sensor data integrity attacks. *IEEE Transactions on Industrial Informatics*.

Yang, B., Guo, L., Li, F., Ye, J., & Song, W. (2020a). Vulnerability assessments of electric drive systems due to sensor data integrity attacks. *IEEE Transactions on Industrial Informatics*, *16*(5), 3301–3310. https://doi.org/10.1109/TII.2019.2948056

Yang, B., Guo, L., Li, F., Ye, J., & Song, W. (2020b). Vulnerability assessments of electric drive systems due to sensor data integrity attacks. *IEEE Transactions on Industrial Informatics*, *16*(5), 3301–3310. https://doi.org/10.1109/TII.2019.2948056

Yang, B., Guo, L., & Ye, J. (2021a). Physics-based attack detection for traction motor drives in electric vehicles using random forest. *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 849–854. https://doi.org/10.1109/APEC42165.2021.9487247

Yang, B., Guo, L., & Ye, J. (2021b). Physics-based attack detection for traction motor drives in electric vehicles using random forest. *2021 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 849–854. https://doi.org/10.1109/APEC42165.2021.9487247

Yang, B., & Ye, J. (2022). Data-driven detection of physical faults and cyber attacks in dual-motor ev powertrains. *2022 IEEE Transportation Electrification Conference & Expo (ITEC)*, 991–996. https://doi.org/10.1109/ITEC53557.2022.9814017

Yang, B., Ye, J., Coshatt, S., Song, W., & Zahiri, F. (2022). Data-driven approach for detection of physical faults and cyber attacks in manufacturing motor drives. *2022 IEEE Energy Conversion Congress and Exposition (ECCE)*, 1–6. https://doi.org/10.1109/ECCE50734.2022.9948143

Yang, B., Ye, J., & Guo, L. (2021). Fast detection for cyber threats in electric vehicle traction motor drives. *IEEE Transactions on Transportation Electrification*, 1–1. https://doi.org/10.1109/TTE.2021.3102452

Yang, B., Ye, J., & Guo, L. (2022). Fast detection for cyber threats in electric vehicle traction motor drives. *IEEE Transactions on Transportation Electrification*, *8*(1), 767–777. https://doi.org/10.1109/TTE.2021.3102452

Ye, H., Yang, K., Ge, H., Magne, P., & Emadi, A. (2014). A drive cycle based electro-thermal analysis of traction inverters. *2014 IEEE Transportation Electrification Conference and Expo (ITEC)*, 1–6. https://doi.org/10.1109/ITEC.2014.6861761

Ye, J., Yang, K., Ye, H., & Emadi, A. (2017). A fast electro-thermal model of traction inverters for electrified vehicles. *IEEE Transactions on Power Electronics*, *32*(5), 3920–3934. https://doi.org/10.1109/TPEL.2016.2585526

Ye, J., Giani, A., Elasser, A., Mazumder, S. K., Farnell, C., Mantooth, H. A., Kim, T., Liu, J., Chen, B., Seo, G.-S., Song, W., Greidanus, M. D. R., Sahoo, S., Blaabjerg, F., Zhang, J., Guo, L., Ahn, B., Shadmand, M. B., Gajanur, N. R., & Abbaszada, M. A. (2022). A review of cyber-physical security for photovoltaic systems. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *10*(4), 4879–4901. https://doi.org/10.1109/JESTPE.2021.3111728

Ye, J., Guo, L., Yang, B., Li, F., Du, L., Guan, L., & Song, W. (2020). Cyber–physical security of powertrain systems in modern electric vehicles: Vulnerabilities, challenges, and future visions. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *9*(4), 4639–4657.

Ye, J., Guo, L., Yang, B., Li, F., Du, L., Guan, L., & Song, W. (2021). Cyber-physical security of powertrain systems in modern electric vehicles: Vulnerabilities, challenges, and future visions. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *9*(4), 4639–4657. https://doi.org/10.1109/JESTPE.2020.3045667

Yulia, C., Pete, B., Andrew, B., Peter, E., Kevin, J., Hugh, S., & Kristan, S. (2016). A review of cyber security risk assessment methods for scada systems. *Computers & Security*, *56*, 1–27.

Zetter, K. (2011). How digital detectives deciphered stuxnet, the most menacing malware in history. *Wired Magazine*, *11*, 1–8.

Zetter, K. (2015). A cyberattack has caused confirmed physical damage for the second time ever.

Zhang, D., Feng, G., Shi, Y., & Srinivasan, D. (2021). Physical safety and cyber security analysis of multi-agent systems: A survey of recent advances. *IEEE/CAA Journal of Automatica Sinica*, *8*(2), 319–333. https://doi.org/10.1109/JAS.2021.1003820

Zhang, J., Guo, L., & Ye, J. (2022). Cyber-attack detection for pv farms based on power-electronics-enabled harmonic state space modeling. *IEEE Transactions on Smart Grid*, *13*(5), 3929–3942. https://doi.org/10.1109/TSG.2021.3121009

Zhang, J., Ye, J., & Guo, L. (2021). Model-based cyber-attack detection for voltage source converters in island microgrids. *2021 IEEE Energy Conversion Congress and Exposition (ECCE)*, 1413–1418. https://doi.org/10.1109/ECCE47101.2021.9595899

Zhang, S., Wang, B., Kanemaru, M., Lin, C., Liu, D., Miyoshi, M., Teo, K. H., & Habetler, T. G. (2020). Model-based analysis and quantification of bearing faults in induction machines. *IEEE Transactions on Industry Applications*, *56*(3), 2158–2170. https://doi.org/10.1109/TIA.2020.2979383

Zhang, T., Antunes, H., & Aggarwal, S. (2014). Defending connected vehicles against malware: Challenges and a solution framework. *IEEE Internet of Things Journal*, *1*(1), 10–21. https://doi.org/10.1109/JIOT.2014.2302386

Zhang, Y., & Yan, J. (2019). Domain-adversarial transfer learning for robust intrusion detection in the smart grid. *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 1–6. https://doi.org/10.1109/SmartGridComm.2019.8909793

Zhou, L., Ouyang, X., Ying, H., Han, L., Cheng, Y., & Zhang, T. (2018). Cyber-attack classification in smart grid via deep neural network. *Proceedings of the 2nd International Conference on Computer Science and Application Engineering*, 90.

Zhu, B., & Sastry, S. (2010). Scada-specific intrusion detection/prevention systems: A survey and taxonomy. *Proceedings of the 1st workshop on secure control systems (SCS)*, *11*, 7.