



Data-Driven Stability Analysis of IBR-rich Grids with hybrid EMT + Phasor Models

Final Project Report

S-106

Power Systems Engineering Research Center
*Empowering Minds to Engineer
the Future Electric Energy System*



Data-Driven Stability Analysis of IBR-rich Grids with hybrid EMT + Phasor Models

Final Project Report

Project Team

Amarsagar Reddy Ramapuram Matavalam, Project Leader

Vijay Vittal

Arizona State University

Subhonmesh Bose

University of Illinois, Urbana-Champaign

Graduate Students

Anirudh Chatti Venkata Sesha

Arnav Bagga

Giritharan Vijay Iswaran

Arizona State University

Boya Hou

University of Illinois, Urbana-Champaign

PSERC Publication 25-05

September 2025

For information about this project, contact:

Amarsagar Reddy Ramapuram Matavalam
Arizona State University
School of Electrical, Computer, and Energy Engineering
Tempe, AZ, 85287
Email: amar.sagar@asu.edu

Power Systems Engineering Research Center

The Power Systems Engineering Research Center (PSERC) is a multi-university Center conducting research on challenges facing the electric power industry and educating the next generation of power engineers. More information about PSERC can be found at the Center's website: <http://www.pserc.org>.

For additional information, contact:

Power Systems Engineering Research Center
Arizona State University
527 Engineering Research Center
Tempe, Arizona 85287-5706
Phone: 480-965-1643
Fax: 480-727-2052

Notice Concerning Copyright Material

PSERC members are given permission to copy without fee all or part of this publication for internal use if appropriate attribution is given to this document as the source material. This report is available for downloading from the PSERC website.

© 2025 Arizona State University. All rights reserved.

Acknowledgements

We would like to express our gratitude to the project advisors for their technical feedback and insightful suggestions, particularly:

- Parag Mitra, EPRI
- Deepak Ramasubramanian, EPRI
- W. Baker, EPRI
- Chetan Mishra, Dominion
- James D.L. Ree, Dominion
- Andrea Pinceti, Dominion
- Katelynn Vance, Dominion
- Harvey Scribner, SPP
- Bo Gong, SRP
- Rishabh Jain, NREL
- S. Akinbode, MISO
- R. Rajasekhar, MISO
- M. Pamperin, MISO
- P. Dalton, MISO
- Manu Parashar, GE
- Kwok Cheung, GE
- Mohit Singh, ComEd

Executive Summary

The large-scale integration of renewable variable generation and inverter-based resources (IBRs) significantly alters system responses to contingencies and fault events, affecting the system's ability to maintain stable operation under critical conditions. Insights from MISO's Renewable Integration Impact Assessment (RIIA) report indicate that when IBR penetration exceeds 30%, voltage and converter-driven stability concerns become prominent. This underscores the critical need to accurately model IBRs in stability studies to ensure a comprehensive understanding of system behavior under high renewable penetration.

With increasing levels of IBR integration, large disturbance stability analysis becomes essential for secure grid operation. Recent events have demonstrated the limitations of phasor domain simulations, emphasizing the need to incorporate full scale electromagnetic transient (EMT) simulations. This presents a major challenge for utilities and ISOs because IBRs are software defined grid components with much faster dynamics, operating on EMT time scales, compared to synchronous generators. In addition, capturing the detailed grid support functionalities of IBRs within full EMT models in large utility scale studies is both computationally demanding and time consuming.

To improve efficiency in stability studies, utilities are increasingly adopting hybrid EMT and phasor models, where IBRs are represented in the EMT domain while the remainder of the grid is modeled in the phasor domain. Although this approach reduces the computational burden, it lacks a unified framework for stability assessment. As a result, while these models can generate valuable data under transient conditions, there remains a pressing need for reliable methods to analyze the resulting multi-timescale data and ensure trustworthy stability evaluations.

To address these challenges, this work develops a data-driven transient stability assessment framework for IBR-rich grids represented by hybrid models. The approach integrates machine learning with fundamental system dynamics to characterize stability performance and localize instability causes during both planning and operations. By constructing a linear representation of the system in a lifted higher-dimensional space, stability analysis can be performed without explicit system equations, enabling scalable and model-agnostic assessment.

A key strength of the framework is its reliance on data rather than detailed vendor models, making it well suited for IBR-dominated systems where components are often provided only as black-boxes. Leveraging black-box simulators, the methodology enables effective stability evaluation and extends naturally to hybrid co-simulation models, underscoring its versatility and relevance for modern grids. Its efficacy is validated through power system case studies across multiple simulation platforms.

This work demonstrates that a data-driven Koopman framework can transform how operators and planners approach transient stability analysis in IBR-rich grids. By learning stability boundaries directly from simulation data, the method reduces reliance on detailed models and avoids the computational burden of brute-force time-domain studies. Once trained, the Koopman operator provides rapid and scalable estimates of critical clearing times across diverse contingencies, enabling broader and more timely security assessments. The implication for industry is a practical

tool that supports large-scale contingency screening and near real-time decision making as renewable penetration deepens. Looking ahead, the Koopman framework can also be extended to voltage stability, providing a unified data-driven approach to assessing both angle and voltage security in modern power systems.

Project Publications:

- [1] A. R. Matavalam, B. Hou, H. Choi, S. Bose, and U. Vaidya, “Data-driven transient stability analysis using the Koopman operator,” *International Journal of Electrical Power & Energy Systems* 162 (2024): 110307.

Project Publications Under Preparation:

- [1] C. Anirudh, A. R. Matavalam, V. Vittal, B. Hou, and S. Bose, “Koopman Operator-Based Data-Driven Transient Stability Analysis with Hybrid EMT–Phasor Models”

Student Theses:

- [1] Boya Hou (2024), *Nonparametric sparse learning of dynamical systems*. Details available at <https://www.ideals.illinois.edu/items/134403>.

Table of Contents

1. Introduction.....	1
1.1 Overview	1
1.2 Overview of the Problem.....	1
1.3 Objective of the Work	3
1.4 Report Organization	4
2. Background.....	5
2.1 Introduction	5
2.1.1 Literature Survey.....	6
2.2 Transient stability assessment	7
2.2.1 Transient stability assessment with Koopman operator.....	9
2.3 Connection between Koopman eigenfunctions and time-domain simulation.....	11
2.4 Learning the indicator function via the Kernel method	14
3. Research Methodology	18
3.1 Introduction	18
3.2 Generator Modeling.....	18
3.2.1 GENCLS Model.....	19
3.2.2 Model 2.2	20
3.2.3 Governor Model	23
3.2.4 Exciter Model.....	23
3.2.5 Power System Stabilizer (PSS)	25
3.3 IBR Modeling for BPS Studies	25
3.3.1 Grid-following inverter control.....	26
3.4 Hybrid EMT+Phasor models.....	31
3.5 Data generation for study	32
4. Results and Discussion	38
4.1 The Single-Machine Infinite-Bus (SMIB) System.....	38
4.2 The Two-Machine Infinite-Bus (TMIB) System	40
4.3 The Ten-Machine 39-Bus New-England System without IBRs	42
4.4 The Ten-Machine 39-Bus New-England System with IBRs	45
4.4.1 Balanced Fault Analysis.....	48
4.4.2 Unbalanced Fault Analysis.....	53

4.5	240-bus WECC test system	55
4.6	Practical insights and Computation burden	58
5.	Conclusions	60
5.1	Future Work	60
	References	62

List of Figures

Figure 1.1 Power capacity projection according to US Energy Information Administration.	1
Figure 1.2 Voltage response with phasor (blue) and EMT (red) models.....	2
Figure 1.3 Proposed framework for data-driven stability assessment of hybrid models.....	4
Figure 2.1 Transient stability analysis framework.....	5
Figure 2.2 Graphical presentation of ROA (ROA is in grey color).....	8
Figure 2.3 Nonlinear evolution of state in the finite-dimensional state space is lifted to a linear evolution of observables in the infinite-dimensional space of all L2-measurable functions of the state.	9
Figure 2.4 Conceptual plot of the Voronoi cells for the various points in D of a 2-D system. The time-averaged value at these points for a well-chosen ψ should be binary that take different values inside and outside the ROA. The two different values are represented by blue and red colors.....	13
Figure 3.1 Typical interface of machine models with the power network.	19
Figure 3.2 Equivalent circuit representation of GENCLS.	19
Figure 3.3 Equivalent circuit in d-axis.	20
Figure 3.4 Equivalent circuit in q-axis.	20
Figure 3.5 Mathematical representation of the IEEEG1 turbine governor model.....	23
Figure 3.6 Type ST1A exciter model.	24
Figure 3.7 General schematic of power system stabilizer (PSS1A).	24
Figure 3.8 Typical IBR plant layout.	26
Figure 3.9 PQ control strategy of GFL.	26
Figure 3.10 Detailed IBR control in PowerFactory.	28
Figure 3.11 Schematic of VSC control in PowerFactory.	28
Figure 3.12 Voltage ride-through chart.	29
Figure 3.13 Decoupling network to obtain the sequence components.	30
Figure 3.14 General schematic of PLL.	30
Figure 3.15 Bode response of PLL considered for the study.....	30
Figure 3.16 Schematic of co-simulation domains.....	31
Figure 3.17 Variation of the exponential term for stable and unstable rotor angle trajectories. ..	34
Figure 3.18 Flowchart of the automated simulation and data processing workflow.	35
Figure 3.19 Expected response of the indicator function.	35
Figure 3.20 Flowchart illustrating the procedure for validating CCT.	36

Figure 4.1 SMIB system connected to an infinite bus through a transmission line.....	38
Figure 4.2 A plot of the estimated indicator function which reveals the ROA.	39
Figure 4.3 Value of the indicator function evaluated at fault-on trajectories with choices of P_e	39
Figure 4.4 TMIB system consists of two generators at buses 1 and 2 connected to the infinite bus at bus 3.	40
Figure 4.5 The value of the indicator function evaluated along the TMIB fault-on trajectory with $E_1 = E_2 = 0.4$	41
Figure 4.6 The total energy and the potential energy of the TMIB faulted system with $E_1 = E_2 = 0.4$. The CCT is estimated to be 0.32s.	41
Figure 4.7 Evolution of generator frequencies in the TMIB system with a fault cleared at (a) $t = 0.285$ s and (b) $t = 0.29$ s.	42
Figure 4.8 The ten-machine New-England Power System (also known as the IEEE 39-bus system).	43
Figure 4.9 The generator ω trajectories for fault scenarios listed in Table 4.3.	44
Figure 4.10 The value of the time-averaged indicator function evaluated along fault-on trajectories for faults listed in Table 4.3.	44
Figure 4.11 IBRs layout in PowerFactory.	46
Figure 4.12 Generator trajectories for Case B and C when cleared at estimated CCT.....	48
Figure 4.13 Generator trajectories for the faults listed in Table 4.5.	50
Figure 4.14 Indicator function response by incorporating moving average method.	51
Figure 4.15 Generator trajectories for the faults listed in Table 4.6.	52
Figure 4.16 Generator trajectories for single phase faults.	53
Figure 4.17 Full view of 240-bus WECC system.	55
Figure 4.18 The indicator functions obtained for the 240-bus WECC system.	57
Figure 4.19 Generator trajectories for Case A through C faults listed in Table 4.9.	57

List of Tables

Table 3.1 Parameters of the synchronous machine.....	21
Table 3.2 VRT settings considered for the study.....	29
Table 4.1 CCT [s] obtained using various for SMIB system.....	40
Table 4.2 Computation times [s] in the different phases for the NE system	43
Table 4.3 Fault information of sample cases studied for the NE system.....	43
Table 4.4 Fault information of the test cases studied for 39 bus system	48
Table 4.5 Estimated CCT using wKNN for three phase faults studied for 39 bus system	49
Table 4.6 Estimated CCT for two phase faults studied for 39 bus system	51
Table 4.7 Estimated CCT for single phase faults studied for 39 bus system.....	54
Table 4.8 Estimated CCT for all fault types using single wKNN model	54
Table 4.9 Estimated CCT for 240-bus WECC system	56
Table 4.10 Summary of simulation studies conducted across different test systems	58

1. Introduction

1.1 Overview

The rapid integration of inverter-based resources (IBRs) into bulk power systems (BPS) has raised significant concerns for transmission system operators and planners, particularly regarding system stability and reliability. IBRs, while capable of enhancing overall system stability, fundamentally alter the dynamic response of the grid to faults and contingencies, thereby affecting its ability to maintain stability across diverse operating conditions. IBRs are required to provide support during disturbances, with their fast-acting dynamics occurring at the electro-magnetic (EMT) timescale, thereby influencing systems transient stability. Capturing the system-wide stability impacts of IBRs requires detailed modeling and simulation of IBR-dominant grids. To address this, the project employs hybrid models for grid simulation and applies data-driven techniques to characterize the stability of modern power grids.

1.2 Overview of the Problem

The power system infrastructure has evolved significantly over the past decades and has undergone significant reformation for incorporating renewable resources, specifically solar- and wind-driven IBRs. According to US Energy Information Administration [1], a total power installation of 48 GW of solar and 17 GW of wind capacity is expected to be installed by the end of 2026 (see Figure 1.1 taken from [1]). Concurrently, the conventional generation is projected to decline by 10 GW. With these ongoing changes in the generation mix, coupled with increasing load demand, power systems are bound to operate near to their stability limits. This trend presents critical challenges for system operators and planners, necessitating a deeper understanding of the stability implications introduced by IBRs. To address these challenges, several working groups and task forces have been established to advance the modeling and simulation of IBR-integrated systems and to assess their impact on system stability across multiple levels.

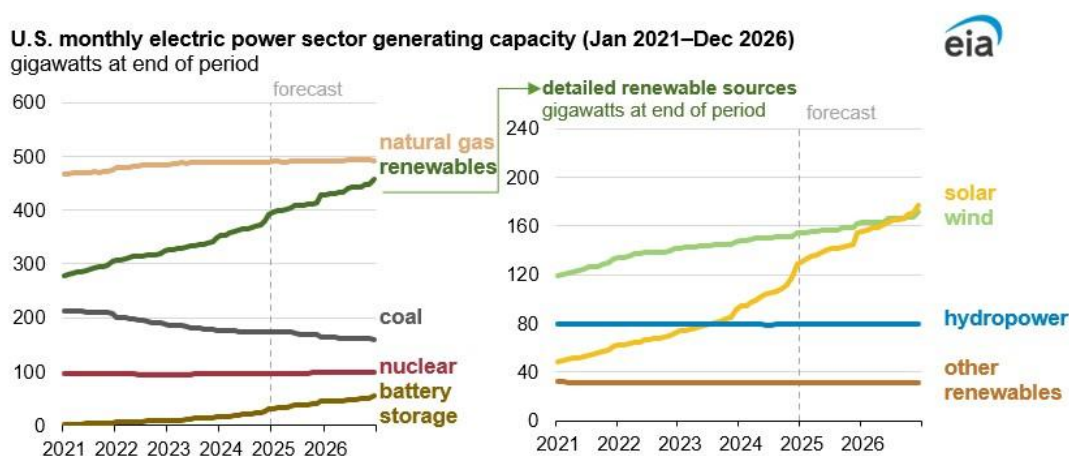


Figure 1.1 Power capacity projection according to US Energy Information Administration.

IBRs present a significant challenge for ISOs and utilities, as they disrupt traditional operational and planning assumptions built around synchronous generators. As the power from IBRs increases, there will be more voltage and converter-driven instabilities. The Renewable Integration Impact Assessment (RIIA) report by MISO [2] indicates that the dynamics of inverter-based resources (IBRs) significantly alter overall grid behavior. As penetration levels approach 35%, the system begins to experience converter-driven instability. IBRs can introduce undesirable dynamics such as subsynchronous oscillations (SSOs), overcurrents, and overvoltages, which have the potential to compromise system stability and damage grid components [3]. For instance, the 2017 event in Texas reported an SSO of 20-30 Hz range lasting 150 ms, resulting in damage to series capacitors and crowbar circuits of the wind turbine [3].

Most of the recent grid events reported in the literature clearly demonstrated the need for large-disturbance stability analysis of IBR-rich grids [3]. In traditional power systems, the dynamics of synchronous generators typically evolved over timescales of hundreds of milliseconds to several seconds due to the system's high inertia. As a result, conventional phasor-domain simulations were generally sufficient to capture the essential dynamics of the power grid. On the other hand, IBR dynamics are fast acting in a timescale of μs and require EMT level modeling and simulation to capture the stability effects of such converter driven systems. This is perhaps the key challenge the utilities and ISOs face are accounting for EMT level simulation for the modern grid. A report by AEMO clearly demonstrated that there is a significantly different response between phasor and EMT simulations for IBR equipped grid under transient conditions (see Figure 1.2 [6]). Following a fault event, phasor-domain simulations indicated a stable voltage response, whereas EMT simulations revealed clear signs of instability. Additionally, IBR-driven disturbances observed in California and Texas have highlighted the limitations of phasor-based models in accurately capturing fast transient behaviors.

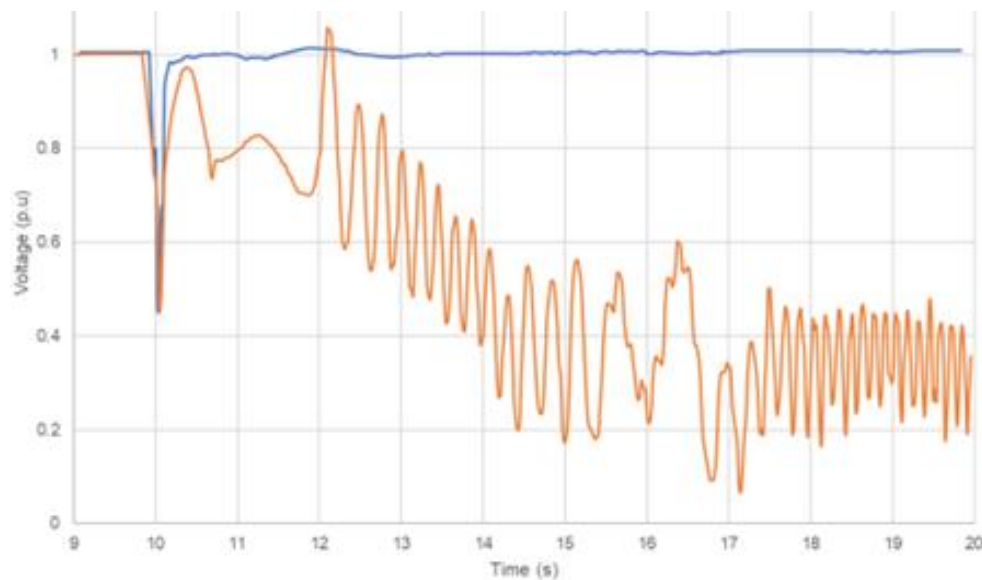


Figure 1.2 Voltage response with phasor (blue) and EMT (red) models.

EMT simulation studies are widely employed for analyzing harmonics, transients, lightning protection, unbalanced conditions, and other fast-acting phenomena. EMT studies have become vital to have reliable BPS operation due to the increasing IBR penetration levels [7]. Some of the important aspects these studies offer include IBR plant-to-plant control interactions, plant-to-grid control interactions, IBRs control stability to fault disturbances, IBR short-circuit analysis, voltage ride-through capability settings impact and protection system behavior with varying IBR penetration levels. Numerous commercial software tools have been developed to support these applications, and EMT-based analysis remains an active and evolving area of research in modern power systems. Consequently, industries and utilities are leaning towards EMT models for IBRs and NERC also recommended that “EMT models should be provided for all newly interconnecting inverter-based resources and EMT studies should be conducted to ensure that performance requirements (once established) are being sufficiently met”.

1.3 Objective of the Work

The incorporation of EMT models into stability studies has become pivotal for understanding the impacts of increasing levels of IBRs. The EMT simulations are computationally exhaustive and thus utilities are keen on using “hybrid models” for stability assessment. Hybrid models refer to co-simulation setup that combines EMT and phasor domain modeling. These are basically used to balance the accuracy and computational burden by applying EMT model where and when needed, and phasor domain for the rest of the system. The IBRs are represented in EMT domain as accurate capture of the resulting dynamics is required, and the rest of the grid is modeled in unbalanced RMS domain. In the co-simulation framework, certain boundaries are defined for EMT and phasor domain, and the desired data exchange takes place across these boundaries. The data can be voltages, currents, and control elements status. Thus, using this framework one can have high resolution analysis without compromising the efficiency of the simulations.

A primary challenge associated with hybrid models is the absence of a unified framework for conventional power system analysis. While these models are effective for simulating the grid under various transient conditions and generating detailed datasets, new methodologies are required to interpret and analyze this multi-timescale data effectively. A second major challenge arising from the rapid integration of IBRs is the diminishing reliability of operators and planners’ intuition in identifying critical regions and operating conditions. This intuition has traditionally played a vital role in guiding efficient computational analysis and developing targeted solutions to mitigate dynamic performance issues.

To address these challenges, the project develops a data-driven approach that integrates machine learning techniques with power system dynamic properties to evaluate the stability performance of hybrid, IBR-rich grid models in both planning and operational stages. The objective of the work is three-fold:

1. Localize the cause of instabilities during transients.
2. Identify key interfaces and IBRs in the grid as the operating/control parameter varies.
3. Identify the limits on transfer flows and IBR injection for secure grid operation.

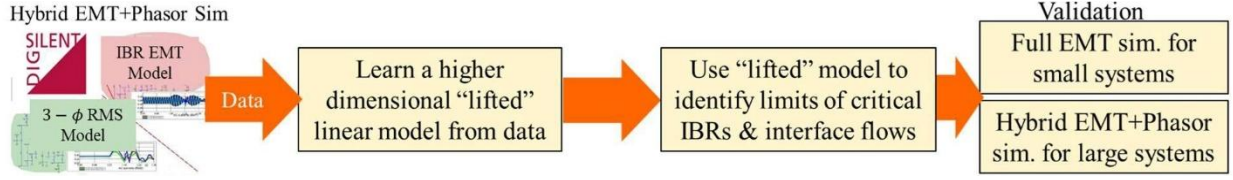


Figure 1.3 Proposed framework for data-driven stability assessment of hybrid models.

The framework for proposed data-driven stability assessment of hybrid EMT+Phasor models is illustrated in Figure 1.3. Rigorous simulations within a co-simulation framework are conducted to generate the data required for stability analysis. This data is then used to construct a linear representation of the system dynamics in a higher-dimensional space through Koopman Operator, ensuring validity across a broad range of operating conditions. The major advantage of the proposed approach is that one can translate the nonlinear dynamics of the power system into a linear model, where one can understand stability, control interactions, and parameter sensitivity of the nonlinear model using familiar linear stability tools. The developed methodology can be extended for stability analysis, investigate control interactions, and enable surrogate model development, among other applications. For clarity and focus, this work concentrates on transient stability assessment in IBR-rich grids.

1.4 Report Organization

The report is organized into five chapters. Chapter 2 presents the theoretical background of transient stability assessment, including an overview of Koopman operator theory and its application to stability studies. Chapter 3 details the modeling of IBRs, synchronous generators, and the adopted co-simulation framework. Chapter 4 discusses the results and insights obtained from various test scenarios. Finally, Chapter 5 summarizes the conclusions and outlines potential directions for future work.

2. Background

2.1 Introduction

Transient stability analysis of power systems primarily focuses on determining whether the system dynamics return to a stable equilibrium point following a disturbance [9], [10],[11]. Such disturbances may include bus or line faults, line outages, or generator outages. In short, this analysis is concerned with whether the system remains in synchronism following a disturbance. Time-domain simulations [11] are the most widely used choice for this analysis where the power system models are solved via numerical integration. Essentially, one simulates the post-fault dynamics starting from various points on the fault-on trajectory to certify whether the system will quickly converge to a stable equilibrium point, exhibit large oscillations, or even diverge. However, accuracy of this method greatly relies on the power system models, and it becomes challenging with the rapid integration of the IBRs. Additionally, the numerical integration of the differential equations is computationally intensive thus rendering it unsuitable for stability assessment with wide range of operating points.

The framework for transient stability analysis is illustrated in Figure 2.1 and comprises of model construction and stability assessment methods. Modeling involves the mathematical representation of power system components, including generators and IBRs, to capture their transient behavior. For example, the classical transient stability models for generators represent the dynamics of generator speed and angle. These dynamics are typically governed by first-order differential-algebraic equations.

Both Lyapunov-based and non-Lyapunov-based methods are widely used for stability assessment. Energy functions, sum-of-squares techniques, and connective stability approaches fall under Lyapunov-based methods. On the other hand, non-Lyapunov-based methods include time-domain simulations, the extended equal-area criterion, and data-driven approaches. Even today, a wide range of methods continues to be explored in this area. Ultimately, it all comes down to the trade-off between the conservativeness of stability estimates, the speed of execution and scalability of the approach to modern power grids.

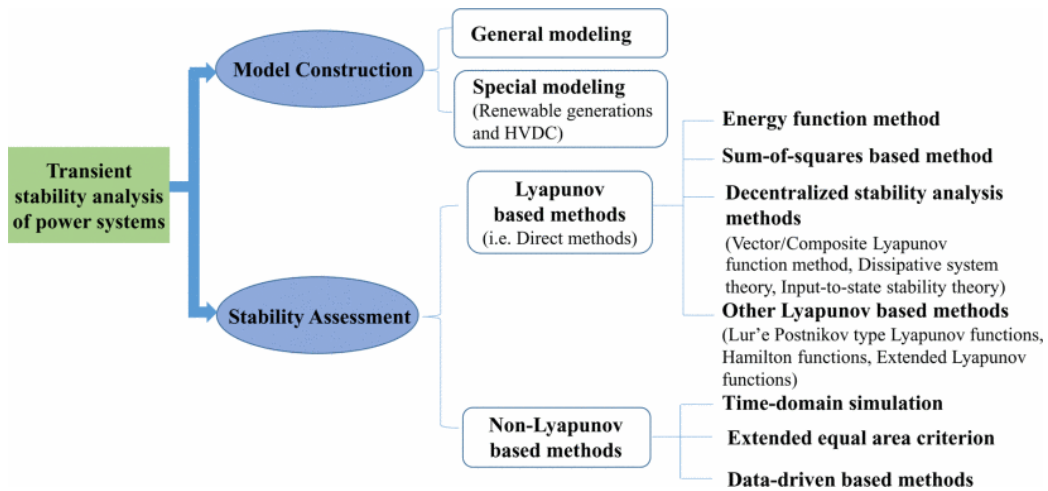


Figure 2.1 Transient stability analysis framework.

2.1.1 Literature Survey

The direct method is an alternative approach to assessing transient stability without explicitly integrating differential equations of post-fault systems. It utilizes Lyapunov-type functions to assess the stability of a post-fault system, based on an approximate stability boundary. Transient energy functions (TEF) expounded in [12]-[15] are examples of such Lyapunov-style functions that decrease along the system trajectory once the fault is cleared [16][17], a critical value of TEF (called the critical energy) is then computed, such that the sublevel sets can be certified to lie within the region of attraction (ROA) of the stable equilibrium point of the post-fault dynamics. To compute such critical energy, the controlling unstable equilibrium point (UEP), closest UEP [18]-[21] and the boundary of stability region based controlling unstable equilibrium point (BCU) method in [13], [22] are widely established. Nevertheless, computing critical energy is not easy, as it requires finding several unstable equilibrium points surrounding the stable equilibrium point of post-fault dynamics and determining which UEP is the most relevant to the faulted system. The Potential Energy Boundary Surface (PEBS) method [23] avoids computing the relevant UEP by finding a local approximation of the stability boundary of the original system model using an analytically derived approximate potential energy function of the power system. However, beyond the single-machine infinite bus system, sufficient conditions for PEBS method to provide accurate stability assessments are not well-understood.

Another major limitation of the direct method is the choice of the Lyapunov function itself, which depends on the mathematical description of the power system model. These functions are only known for a limited class of simplified power system models. Authors in [24][27] have proposed to construct such functions for general power systems using polynomial optimization, following the seminal work of Parrilo in [17]. While promising in theory, state-of-the-art software for these methods does not scale favorably with the size of the model. In particular, they require explicit analytical models of various power system components. This poses a challenge since deriving accurate system equations that capture nuances of modern power system components ex-ante is not only tedious but may be impossible in some situations. For example, consider the emerging inverter-dominated power systems where the inverter/plant dynamics are not expressed as analytical equations, but are instead represented by compiled dynamically linked libraries to protect the intellectual property of inverter manufacturers. In other words, the underlying dynamics of an inverter are deliberately hidden from system operators to protect intellectual property, thus limiting the applicability of the direct method. Even if the mathematical models of these components were easily accessible, the exact representation of the model, whether in the phasor domain or a more detailed counterpart, can heavily influence stability assessments, e.g., see [28]. Furthermore, the aforementioned methods are all fault-dependent and focus on assessing a particular fault. Since the critical energy is different for each fault, the computational challenge compounds with multiple possible counterfactual fault scenarios.

Recent works have applied a variety of data driven methods to transient stability assessment and critical clearing time prediction. Authors proposed LSTM based framework using synchrophasor data for online stability assessment in conventional systems for conventional power grids [29]. The critical clearing time prediction using cost sensitive data-driven scheme is reported in [30], where the algorithm is model agnostic and not fully tested with IBRs. Other studies explored machine learning and ensemble techniques on IEEE benchmark systems, primarily without IBRs, to predict stability outcomes and CCT [31]-[33]. In addition, Mohammed et al. [34] and Wang et

al. [35] proposed a data driven approach for estimating the impedance of IBRs to enable more efficient stability evaluation, further highlighting the growing role of data oriented methods in modern power system analysis.

The need for scalable techniques arises because utilities and ISOs must screen thousands of contingencies across wide operating conditions in near real-time, particularly as IBR penetration increases and their opaque, vendor-specific controls introduce fast dynamics that are not captured by traditional phasor-domain tools. Brute-force EMT or hybrid EMT–phasor simulations provide valuable insights, but their computational demands make them impractical for system-wide security assessment.

Thus, the gap in the literature is clear: while data-driven transient stability assessment methods exist, there is no scalable, model-agnostic framework that can leverage only black-box simulation or measurement data to provide reliable stability assessment in IBR-rich systems. Addressing this requires extending data-driven operator-theoretic methods, such as Koopman lifting, to deliver both computational efficiency and robustness in the presence of opaque IBR dynamics.

To cope with these challenges, a data-driven transient stability assessment based on the Koopman operator is developed in this work. First developed in [36], the Koopman operator lifts the finite-dimensional nonlinear evolution of the state to a linear but infinite-dimensional evolution in the function space of observables (scalar mappings of the state). Approximation of the Koopman operator from data is well studied in the literature, see [37],[42] for details. As a linear operator, its spectra reveal information relevant to stability monitoring [43]-[46]. In this report, we show that eigenfunctions of the Koopman operators of the post-fault dynamics reveal its ROA. One advantage of using the Koopman operator for transient stability analysis is that these eigenfunctions can be learned from data of states propagated through the system dynamics, but do not require explicit analytical models. Prior literature such as [48][50] have investigated data-driven transient stability assessment by leveraging the learned Koopman operator. In contrast to [33], in this work we make direct use of the Koopman operator for estimating CCT rather than relying on eigenfunctions, with input features deliberately kept limited to enhance efficiency and scalability. Nevertheless, accurate estimates of the Koopman operator and its leading eigenfunctions require a pre-selected set of observables to be rich enough. Given the challenges of selecting such a basis, we leverage the results in [51] to approximate the Koopman eigenfunction nonparametrically in a reproducing kernel Hilbert space (RKHS). As shown in [52], kernel methods demonstrate excellent empirical performance in data-limited regimes. Still, the scalability often suffers with larger datasets. To counter this difficulty, one can allow selective loss in the representation of that operator to control model complexity.

2.2 Transient stability assessment

Transient stability analysis of power systems determines whether the system remains in synchronism after being subjected to disturbance. In a typical transient stability scenario, the system's operation can be divided into three stages: pre-fault, fault-on, and post-fault. The power grid experiences different dynamics during each of these stages. Fundamentally, transient stability is an assessment of the asymptotic stability of the post-fault operating condition. To illustrate this, consider a dynamical system represented by the following set of differential equations:

$$x'(t) = \begin{cases} F_0(x(t)), & t_0 \leq t < t_f \\ F_f(x(t)), & t_f \leq t < t_{fc} \\ F(x(t)), & t_{fc} \leq t < \infty \end{cases} \quad (2.1)$$

In (2.1), $x(t)$ represents the state variables of the system under study, and the times t_0 , t_f and t_{fc} correspond to the pre-fault, fault-on, and fault-clearing instants, respectively. The pre-fault condition refers to the stable operating point of the system, with its dynamics governed by F_0 . During the fault-on period starting at t_f , the system dynamics change and are governed by F_f until the fault is cleared at t_{fc} . Therefore, the fault duration lies within the interval $[t_f, t_{fc})$.

Let us denote the system state trajectory by $\Phi(x, t)$. Then, the state of the system at the fault-clearing instant can be expressed as $\Phi_F(x_0, t_{fc} - t_f)$ where x_0 is the initial state at t_f . Let x_s denote the equilibrium point of post-fault dynamics. The transient stability question is then mathematically formulated in (2.2), which essentially asks: starting from the post-fault initial condition, will the system trajectory converge to the equilibrium point x_s ?

$$\lim_{t \rightarrow \infty} \Phi(x_{fc}, t) = x_s \quad (2.2)$$

Thus, there exists a region of initial conditions in the state-space where trajectories converge to x_s . This region is known as region of attraction (ROA), denoted by $\text{ROA}(x_s)$ (illustrated in Figure 2.2). Consider the ROA of a post-fault system with equilibrium point x_s . If the fault is cleared at the time $x_{c1,a}$, and the system trajectory subsequently converges to x_s , the system is considered transiently stable.

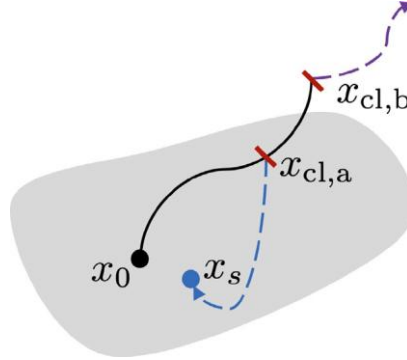


Figure 2.2 Graphical presentation of ROA (ROA is in grey color).

However, if the fault is cleared at a point outside the ROA (e.g., at $x_{c1,b}$), the system trajectory diverges, leading to transient instability. The maximum allowable time for fault clearance that still ensures stability is known as the Critical Clearing Time (CCT). Therefore, transient stability analysis aims to determine the CCT, and fundamentally, the fault duration ($t_{fc} - t_f$) must be less than the CCT to maintain system stability.

2.2.1 Transient stability assessment with Koopman operator

This section presents the relation between CCT and the Koopman eigenfunctions. Consider the indicator function $\text{ROA}(x_s)$, defined as

$$\bar{\varphi}(x) = \begin{cases} 1, & x \in \text{ROA}(x_s) \\ 0, & \text{otherwise} \end{cases} \quad (2.3)$$

To calculate the CCT for a fault originating at x_0 , one then needs to numerically integrate the on-fault trajectory (denoted $x_{\text{on-fault}}(t)$) and evaluate

$$\text{CCT} := \max\{t \geq 0 \mid \bar{\varphi}(x_{\text{on-fault}}(t)) = 1\} \quad (2.4)$$

i.e., the time until which the evaluation of $\bar{\varphi}$ along the fault-on trajectory remains unity before falling off to zero.

We next relate $\bar{\varphi}$ to an eigenfunction of the Koopman operator. Consider a discrete deterministic non-linear dynamical system with states in $\mathbb{X}$, described by

$$x_{t+1} = F(x_t) \quad (2.5)$$

where $x_t \in \mathbb{X}$ and $F: \mathbb{X} \rightarrow \mathbb{X}$ is locally Lipschitz continuous and invertible. For power systems represented by differential-algebraic equations (DAE), F represents the numerical integration of the DAE with a fixed time-step. Let $h: \mathbb{X} \rightarrow \mathbb{C}$ be an observable (i.e., a scalar-valued function) over $\mathbb{X}$. The Koopman operator $\mathcal{K}$ acts on h as

$$[\mathcal{K}h](x) = h(F(x)) \quad (2.6)$$

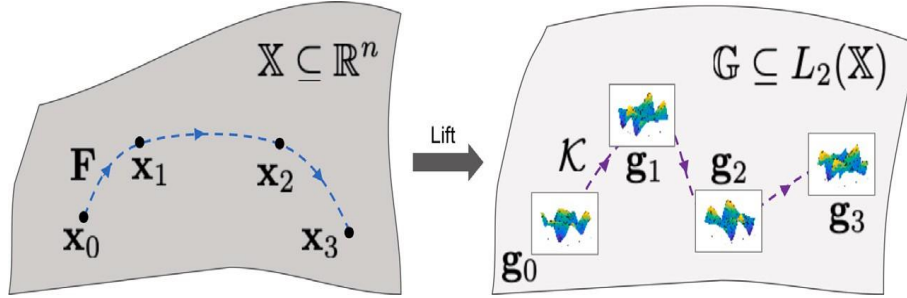


Figure 2.3 Nonlinear evolution of state in the finite-dimensional state space is lifted to a linear evolution of observables in the infinite-dimensional space of all L_2 -measurable functions of the state.

This operator thus maps the observable g to $\mathcal{K}g$, which when evaluated at x , is g evaluated at the state after being pushed forward through the system dynamics in one time-step as Figure 2.3 reveals. This operator is infinite-dimensional as it acts on a suitably defined space of functions of the states. It is easy to verify that

$$\mathcal{K}(\alpha_1 g_1 + \alpha_2 g_2) = \alpha_1 \mathcal{K} g_1 + \alpha_2 \mathcal{K} g_2 \quad (2.7)$$

for scalars α_1, α_2 and observables g_1, g_2 and hence, is a linear operator. The spectrum of this operator is rich in information about the dynamics of the system. As we will demonstrate, the indicator function $\bar{\varphi}$ of $\text{ROA}(x_s)$ is intimately related to the eigenfunctions of $\mathcal{K}$. Before we establish that relationship, we emphasize that the linear operator description of the dynamical system in the space of observables is distinct from a local linearization around a point in the state space.

Let $\varphi_\lambda(x)$ be an eigenfunction of $\mathcal{K}$ with eigen value λ , i.e., $[\mathcal{K}\varphi_\lambda](x) = \lambda\varphi_\lambda(x)$ for all $x \in \mathbb{X}$. In particular if φ_1 is eigenfunction with eigen value equal to 1, then

$$[\mathcal{K}\varphi_1](x) = \varphi_1 F(x) = \varphi_1(x) \quad (2.8)$$

Applying Koopman operator n times, we get

$$\varphi_1(x) = [\mathcal{K}^n \varphi_1](x) = \varphi_1(F^n(x)) \quad (2.9)$$

For any $x \in \text{ROA}(x_s)$, we have $\lim_{n \rightarrow \infty} F^n(x) = x_s$, which combined with (2.8) gives

$$\varphi_1(x) = \varphi_1(\lim_{n \rightarrow \infty} F^n(x)) = \varphi_1(x_s) \quad (2.10)$$

In other words, φ_1 is constant over $\text{ROA}(x_s)$. Said even more differently, φ_1 over $\text{ROA}(x_s)$ is a scaled version of the indicator function $\bar{\varphi}$ of $\text{ROA}(x_s)$. As a result, one can utilize the eigenfunction of the Koopman operator whose eigenvalue is equal to one to construct the indicator function of $\text{ROA}(x_s)$ with which one can then perform transient stability analysis. This analysis then reduces the estimation of such an eigenfunction from data suitably obtained from system trajectories.

We further remark that the theory presented in this section also applies to systems specified by the algebraic states. This is because the information of mapping between the algebraic states and the dynamic states is inherently embedded into the trajectory of the dynamic states [55]. In other words, only the dynamic states need to be sampled and analyzed to identify the ROA of post-fault dynamics and the CCT of a fault.

Concept Overview

The Koopman framework offers a way to analyze nonlinear system dynamics through a linear lens. Instead of working directly with the nonlinear equations of a power system, the method tracks simple functions of the state (called observables) in an expanded space called the lifted space. In this lifted space, the evolution of the system is captured by a linear operator known as the Koopman operator. Although the full operator is infinite-dimensional, in practice it can be approximated using simulation data. This finite approximation allows one to predict how the system behaves under disturbances and to estimate stability boundaries in terms of the critical clearing time. To improve clarity and make the idea more accessible, a simple numerical example is also provided, illustrating step by step how nonlinear dynamics can be recast into a linear representation in the lifted space.

Let us consider a two-dimensional discrete nonlinear system

$$x_1(t+1) = \rho x_1(t) \quad (2.11)$$

$$x_2(t+1) = \mu x_2(t) + (\rho^2 - \mu)c(x_1(t)^2) \quad (2.12)$$

The first equation shows that x_1 evolves linearly with scaling factor ρ , whereas the second equation contains a quadratic nonlinearity in x_1 . To capture this behavior within the Koopman framework, we introduce a lifted state vector (dictionary of observables)

$$\Psi = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \end{bmatrix} \quad (2.13)$$

The idea is that although the original system is nonlinear, the evolution of these observables can be expressed linearly. To demonstrate this, we compute the evolution of each observable. The constant observable remains unchanged, i.e. $1(t+1) = 1$. The first state evolves as $x_1(t+1) = \rho x_1(t)$. The second state evolves as $x_2(t+1) = \mu x_2(t) + (\rho^2 - \mu)c(x_1(t)^2)$, which is a linear combination of x_2 and x_1^2 . Finally, the quadratic observable evolves as $x_1^2(t+1) = \rho^2 x_1^2(t)$.

Each of these updates can be written as a linear combination of the basis functions in $\Psi(x)$. Collecting the coefficients leads to the finite-dimensional Koopman operator K

$$K = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \rho & 0 & 0 \\ 0 & 0 & \mu & (\rho^2 - \mu)c \\ 0 & 0 & 0 & \rho^2 \end{bmatrix} \quad (2.14)$$

Thus, the lifted state satisfies the linear recursion

$$\Psi(t+1) = K\Psi(t) \quad (2.15)$$

In this way, the nonlinear dynamics of the original two-dimensional system (2.11)-(2.12) are exactly represented by a linear operator (K) in the four-dimensional lifted space (Ψ). The advantage for power systems is that stability boundaries such as CCT can be estimated using linear tools rather than repeated nonlinear simulations. This enables faster, scalable, and model-agnostic stability assessment, even in IBR-rich grids.

2.3 Connection between Koopman eigenfunctions and time-domain simulation

The ROA of a stable equilibrium point of the post-fault dynamics has long been estimated using repeated numerical integration of the post-fault dynamical system model, starting from a collection of points around the boundary of the ROA of an equilibrium point. Having studied the relationship between the eigenfunction of the Koopman operator with $\lambda = 1$ and the indicator function of the ROA, it is natural to surmise a connection between these eigenfunctions and the time-domain

simulation approach to transient stability analysis. In this section, we precisely establish said connection. As shall become clear, such a relation sheds light on a computational scheme to estimate the eigenfunction with an eigenvalue equal to 1.

Consider a bounded function $\psi : \mathbb{X} \rightarrow \mathbb{C}$ on the state space with $|\psi| \leq \mathbf{B}$. Then, define the N – step time-averaged version of ψ along a trajectory starting from x as

$$\psi_N(x) := \frac{1}{N} \sum_{n=1}^N \psi(F^n(x)) \quad (2.16)$$

Then, for $x \in \mathbb{X}$, the triangle inequality gives

$$|\psi_N(F(x)) - \psi_N(x)| = \frac{1}{N} |\psi_N(F^{N+1}(x)) - \psi(x)| \leq \frac{2\mathbf{B}}{N}$$

For large N , the above bound suggests that

$$[\mathcal{K}\psi_N](x) = \psi_N F(x) = \psi_N(x) \quad (2.17)$$

This relation, together with (2.8), indicates that ψ_N converges pointwise to the eigenfunction of the Koopman operator whose eigenvalue equals one. In the sequel, we refer to the eigenfunction with $\lambda = 1$ as the unit eigenfunction of the Koopman operator.

The above calculation sheds light on a computational scheme to estimate a unit eigenfunction. Starting with a bounded function ψ , one can estimate ψ_N by averaging ψ along a trajectory over N time-steps obtained via time-domain simulation. Then, this estimated unit eigenfunction should be roughly constant within $\text{ROA}(x_s)$, using which one can then perform transient stability analysis. While this computational scheme does not require explicit evaluation of the Koopman operator, it reveals the link between the conventional time-domain simulation approach to the estimation of $\text{ROA}(x_s)$ and the unit eigenfunction of the Koopman operator. Although Koopman operator learning has been applied to analyze power systems in [48][49], to the best of our knowledge, this paper is the first to draw such a connection between time-domain simulation and Koopman eigenfunctions for transient stability analysis. We conclude this section with remarks on the practical implementation of estimating an eigenfunction via this approach.

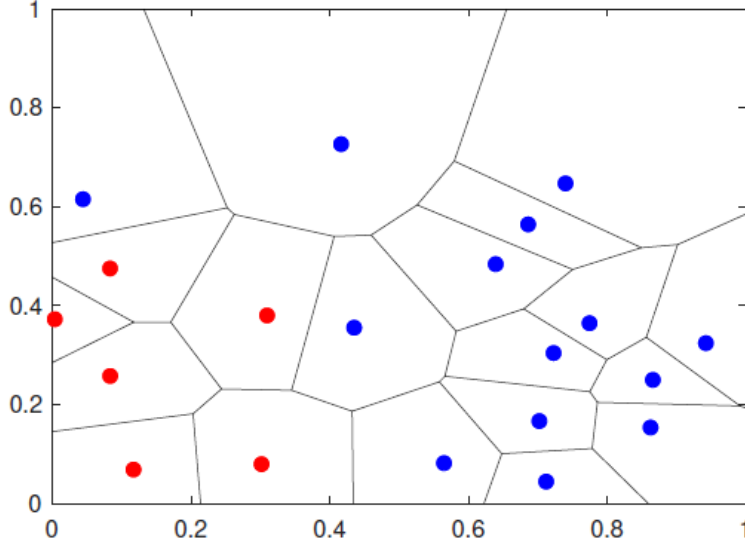


Figure 2.4 Conceptual plot of the Voronoi cells for the various points in $\mathbb{D}$ of a 2-D system. The time-averaged value at these points for a well-chosen ψ should be binary that take different values inside and outside the ROA. The two different values are represented by blue and red colors.

It is time-consuming to estimate Ψ_N everywhere in state space $\mathbb{X}$. Instead, one can probe Ψ_N at a finite number of points $\mathbb{D} := \{x_1, x_2, \dots, x_m\}$, sampled around the portion of the state space “near” the fault-on trajectory, and then interpolate Ψ_N at any point of interest by assigning the value of Ψ_N at its nearest-neighbor in $\mathbb{D}$. Points in $\mathbb{D}$ thus partition the state space $\mathbb{X}$ into separate regions, often referred to as *Voronoi cells*, as illustrated in Figure 2.4, where Ψ_N takes two different values on $\mathbb{D}$. The interpolated value of Ψ_N (call it $\widehat{\Psi}_N$) is then defined by

$$\widehat{\Psi}_N(x) := \Psi_N(\arg \min_{x' \in \mathbb{D}} \|x - x'\|) \quad (2.18)$$

for a suitably defined norm $\|\cdot\|$. Fast open-source implementations for computing the nearest neighbor are readily available, e.g., see [56]. In the sequel, we call this approach the time-averaged approach.

Generally, the choice of ψ will affect the quality of ROA estimation using this approach. As our numerical experiments will suggest, a choice of ψ that is positive everywhere equals one at x_s and that rapidly decays away from x_s tends to yield favorable results. N can be set based on time commitments of time-domain simulation of the post-fault dynamics and where $\Psi_N \approx \Psi_{N-1}$ for all points in $\mathbb{D}$. This time-averaging approach to computing a unit eigenfunction requires collecting post-fault trajectories over a long-time horizon, which can be computationally intensive, and each trajectory is analyzed independently. The approach we present next, on the other hand, utilizes multiple one-step propagation of data points that are utilized together to obtain an interpolation.

2.4 Learning the indicator function via the Kernel method

One can approximate a unit eigenfunction of the Koopman operator in a suitable function space. In this section, we present an approximation technique to find a unit eigenfunction within a real-valued reproducing kernel Hilbert space (RKHS), in which the dataset consists of states propagated one time-step via the system dynamics.

We start by introducing the real-valued RKHS. Consider a symmetric, bivariate, positive definite, continuous, and bounded *kernel* function $\kappa : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}$ that satisfies the following reproducing property for $x \in \mathbb{X}$. Then, define $\mathcal{H}$ to the Hilbert space of functions, given by the completion of the span $\{\kappa(x, \cdot) : x \in \mathbb{X}\}$. In this space, the kernel satisfies the following reproducing property

$$\langle f, \kappa(\cdot, x) \rangle = f(x), \forall f \in \mathcal{H} \quad (2.19)$$

where $\langle \cdot, \cdot \rangle$ computes the inner product of $\mathcal{H}$. Associated with kernel κ its feature map

$$\nu_\kappa(x) = \kappa(x, \cdot) \in \mathcal{H} \quad (2.20)$$

To learn Koopman eigen functions from $\mathcal{H}$, one needs to learn to understand how F interacts with functions $\mathcal{H}$, where F describes the system evolution in (2.5). Assume that $\mathcal{H}$ is *closed* under the system dynamics, i.e.,

$$f \in \mathcal{H} \rightarrow \mathcal{K}f = f \circ F \in \mathcal{H} \quad (2.21)$$

The assumption that $\mathcal{H}$ is closed under system dynamics is common in approximation of linear transfer operators, e.g., see [57][58]. It holds when the Koopman operator $\mathcal{K}$ is a Hilbert Schmidt operator mapping from $\mathcal{H}$ to $\mathcal{H}$. Under this assumption, the Koopman operator admits a closed-form expression using (cross)-covariance operators, per [51],[54], and it can be learned from a collection of data points in $\mathbb{X}$ and their one-step propagation through the system dynamics F . In what follows, we introduce this expression of the operator and present an approach to learn a *sparse* variant of this operator with data from which we then compute a unit eigenfunction for transient stability analysis. We also note that recent works in [60], [61] hold promise to relax the closeness assumption by considering the Koopman operator as a Hilbert–Schmidt operator mapping from $\mathcal{H}$ to the space of square-integrable functions.

The expression for the Koopman operator interacting with an RKHS given in [51], [54] relies on a *stochastic* dynamical system model rather than the deterministic model we introduced in (2.5). Consider a stochastic dynamical system characterized by a transition kernel $p_F(x_{t+1}, x_t)$ that maps the state x_t at time t to a probability density of states at time $t + 1$. The deterministic variant defined by F can be understood as a stochastic kernel that puts all its mass at $F(x_t)$ at time $t + 1$, given the state x_t at time t . With this notation, suppose that $\mathbb{P}(X, X^+)$ denotes a joint distribution over $\mathbb{X} \times \mathbb{X}$, where X is sampled according to a sampling distribution and X^+ is the induced state after X is propagated through the system dynamics p_F . Covariance operators can then be defined as

$$\begin{aligned} C_{XX} &:= \mathbb{E}_X[\nu_\kappa(X) \otimes \nu_\kappa(X)] \\ C_{XX^+} &:= \mathbb{E}_{(X, X^+)}[\nu_\kappa(X) \otimes \nu_\kappa(X^+)] \end{aligned} \quad (2.22)$$

which are elements of the tensor product space $\mathcal{H}_\otimes := \mathcal{H} \otimes \mathcal{H}$, where $\mathbb{E}_{(X, X^+)}$ is expectation with respect to $\mathbb{P}(X, X^+)$. Under a closedness assumption similar to (2.21)—precisely that $\mathbb{E}_{(X, X^+)}[f(X^+)|X] \in \mathcal{H}$ for all $f \in \mathcal{H}$ —it follows from [[61], Theorem 4] that one can define the Koopman operator using the aforementioned covariance operators as

$$\mathcal{K} := C_{XX}^\dagger C_{XX^+} \quad (2.23)$$

See [26] for details. The prevalent definition of Koopman operator interacting with an RKHS is via $\mathcal{K} := C_{XX}^\dagger C_{XX^+}$. However, as pointed out by [62], $C_{XX}^\dagger$ is not globally defined if $\mathcal{H}$ is infinite-dimensional. As such, we consider its regularized variant,

$$\mathcal{K}_\epsilon := (C_{XX} + \epsilon I)^{-1} C_{XX^+} \quad (2.24)$$

where I is the identity operator. Owing to the regularization parameter $\epsilon > 0$, the regularized covariance operator is positive definite and thus admits an inverse.

One can construct the operator $\mathcal{K}_\epsilon$ from data as follows. Consider a data-set $\mathbb{M} := \{(x_1, x_1^+), \dots, (x_m, x_m^+)\}$ of m samples i.i.d. from $\mathbb{P}(X, X^+)$. Then, the empirical estimates of C_{XX} and C_{XX^+} can be computed via sample average as

$$\begin{aligned} \widetilde{C}_{XX} &= \frac{1}{m} \sum_{i=1}^m \nu_\kappa(x_i) \otimes \nu_\kappa(x_i) \\ \widetilde{C}_{XX^+} &= \frac{1}{m} \sum_{i=1}^m \nu_\kappa(x_i) \otimes \nu_\kappa(x_i^+) \end{aligned} \quad (2.25)$$

While the approximation accuracy improves with more data, data storage and computational issues compound when m grows large. To circumvent this burden, we prune $\mathbb{M}$ to construct a sparse dictionary $\mathbb{M}_\gamma$ based on the notion of *coherency* introduced in [63]. Specifically, we construct $\mathbb{M}_\gamma$ by identifying a subset of $\mathbb{M}$ that satisfies

$$\frac{|\kappa_\otimes((x_i, x_i^+), (x_j, x_j^+))|}{\sqrt{\kappa_\otimes((x_i, x_i^+), (x_i, x_i^+)) \kappa_\otimes((x_j, x_j^+), (x_j, x_j^+))}} \leq \gamma \quad (2.26)$$

For each i, j such that $(x_i, x_i^+), (x_j, x_j^+)$ are in $\mathbb{M}_\gamma$. This condition ensures that points that are sufficiently “close” to each other in $\mathbb{M}$ are discarded as they do not add to the richness of the representation of the operators in question. Here, $\kappa_\otimes$ is defined as $\kappa_\otimes((x_1, x_1^+), (x_2, x_2^+)) = \kappa(x_1, x_2) \kappa(x_1^+, x_2^+)$. Let $\mathcal{J}_\gamma$ be the indices among $1, 2, \dots, m$ for which (x_i, x_i^+) are in $\mathbb{M}_\gamma$ and defines the feature matrices V_X, V_{X^+} whose i th columns are feature map ν_κ centered at x_i, x_i^+ for $i \in \mathcal{J}_\gamma$. One can construct the sparse estimator of $C_{XX}(C_{XX^+})$ as

$$\begin{aligned}\widehat{\mathcal{C}_{XX^+}} &= V_X A_\alpha V_{X^+}^T \\ \widehat{\mathcal{C}_{XX}} &= V_X A_\beta V_X^T\end{aligned}\tag{2.27}$$

where A_α (and similarly, A_β) is a diagonal matrix whose entries on the main diagonal minimize

$$\left\| \frac{1}{m} \sum_{i=1}^m v_\kappa(x_i^+) \otimes v_\kappa(x_i) - \sum_{j_\gamma} \alpha_i v_\kappa(x_i^+) \otimes v_\kappa(x_i) \right\|_{\mathcal{H}_\otimes}^2\tag{2.28}$$

over $\alpha \in \mathbb{R}^{j_\gamma}$. The *sparse kernel Koopman estimator* can then be defined as (per [37]).

$$\widehat{\mathcal{K}_\epsilon} := (\widehat{\mathcal{C}_{XX}} + \epsilon I)^{-1} \widehat{\mathcal{C}_{XX^+}}\tag{2.29}$$

When underlying system is ergodic and the stationary distribution is absolutely continuous with respect to the Lebesgue measure on $\mathbb{X}$, [51] further studies finite-sample convergence behavior of such a sparse estimator. We next compute the eigenfunction of $\widehat{\mathcal{K}_\epsilon}$ whose eigen value is 1. Define the Gram matrices as (2.30) and construct Y given in (2.31)

$$G_{XX} = V_X^T V_X, G_{X^+X} = V_{X^+}^T V_X\tag{2.30}$$

$$Y := (A_\beta G_{XX} + \epsilon I)^{-1} A_\alpha G_{X^+X}\tag{2.31}$$

From [40, Proposition 3.1], an operator of the form $\widehat{\mathcal{K}_\epsilon} = V_X Y V_X^T$ an eigenvalue equals to one with the corresponding eigenfunction

$$\varphi_1(x) = k(x)w, [k(x)]_i = \kappa(x_i, x), i \in \mathcal{J}_\gamma\tag{2.32}$$

if and only if w is a right eigenvector of Y associated with the same eigenvalue. This observation enables us to construct eigenfunctions of $\widehat{\mathcal{K}_\epsilon}$ analytically from finite-dimensional Gram matrices G_{XX} and G_{X^+X} .

However, the unit eigenfunction (2.32) is not perfectly constant inside the ROA of the stable equilibrium as it is an approximation derived from a finite representation of the Koopman operator instead of the ideal unit eigenfunction. We thus estimate an indicator function from $\varphi_1(x)$ via

$$\bar{\varphi}_1(x) = \begin{cases} 1, & \text{if } |\varphi_1(x) - \varphi_1(x_s)| \leq \epsilon_o \\ 0, & \text{otherwise} \end{cases}\tag{2.33}$$

where ϵ_o defines a user-specified tolerance. We remark that alternatively, one can also apply clustering techniques to $\varphi_1(x)$ where the number of clusters is two. The RKHS-based approximation to a unit eigenfunction is summarized in Algorithm 1.

Algorithm 1 Approximating Indicator Function in RKHS

Require: kernel κ ; sample pairs $\mathbb{M}$; coherence parameter γ

- 1: Prune $\mathbb{M}$ to $\mathbb{M}_\gamma$ that satisfies (2.26)
 - 2: Solve for the coefficients α, β as in (2.28)
 - 3: Construct Y as in (2.31)
 - 4: Find the right unit eigenvector $\boldsymbol{v}$ of Y and compute unit eigenfunction $\varphi_1(x)$ using (2.32)
 - 5: Compute $\bar{\varphi}_1(x)$ using (2.33)
 - 6: **return** $\bar{\varphi}_1(x)$
-

The proposed methodologies are developed using well-studied implications of the interaction of the Koopman operator with RKHSs in [40], for which sample complexity has been characterized in [51]. In addition, to control the growth of model complexity, we propose to learn a *sparse* variant of the Koopman operator from data using the framework in [51]. We emphasize that thanks to the data-oriented nature of our proposed methods, our algorithm can be used in tandem with black-box simulators such as PSSE, PSLF, PowerWorld, DIgSILENT, etc., and do not rely on the mathematical structure of the power system model. With a learned post-fault dynamical system, our method allows one to analyze multiple fault-on trajectories efficiently.

3. Research Methodology

3.1 Introduction

A power system model represents the electrical network and its components, including generators, transmission lines, transformers, and control systems. It is used to study system behavior under both steady-state and dynamic conditions, including faults. Such modeling is central to stability analysis, providing insights into how the grid responds to disturbances. Accurate models and simulation tools are therefore critical for planners and operators to maintain reliable and secure grid operation. This chapter provides the modeling of generators, IBRs, and associated control design considered for stability assessment. The modeling and simulation are carried out in PSS/E and PowerFactory.

3.2 Generator Modeling

Synchronous machine modeling is a well-established topic in literature. Advances in modern computational capabilities now enable detailed representations of machine dynamics, allowing for more accurate characterization of instability phenomena. A synchronous machine comprises two main components: the stator and the rotor. The stator accommodates the armature windings, while the rotor carries the field (excitation) windings. Of particular interest in stability studies is the accurate capture of rotor angle and speed dynamics, as these variables are fundamental indicators of system synchronism. Accordingly, the generator model characterizes the interactions among these components and their dynamic response under varying operating conditions.

Most software tools such as Digsilent PowerFactory, PowerWorld, and MATLAB/Simulink, adopt the dq -domain approach for generator modeling. At the foundation of all these models lies the swing equation, as expressed in (3.1)

$$\begin{aligned}\frac{d\delta}{dt} &= \omega - \omega_s \\ \frac{d\omega}{dt} &= \frac{1}{2H} (P_{mec} - D\omega - P_{elec})\end{aligned}\tag{3.1}$$

where δ is the generator angle, ω is the generator speed, ω_s is the synchronous speed, H is the inertia constant, D is the damping constant, and P_{elec} is the electric power of the generator. Under stable operating conditions, the system remains in synchronism i.e., $\omega = \omega_s$. During fault conditions, the rotor accelerates ($P_m > P_e$) and decelerates ($P_m < P_e$) until new steady state operation point is established. All generator models inherently include two fundamental states of interest namely δ and ω . These states form the basis of the swing equation, which governs electromechanical dynamics. Consequently, a substantial body of research has focused on analyzing and efficiently solving this equation for large-scale power systems. Higher order models additional states such as flux linkages and damper winding dynamics; nonetheless, δ and ω remain central states across all the synchronous machine models. In the literature, several synchronous machine models are presented, each differing in the level of detail considered. Examples include GENCLS, GENROU, GENSAL, among others. This study employs the GENCLS and Model 2.2 synchronous machine models, with their governing equations presented in the following sections.

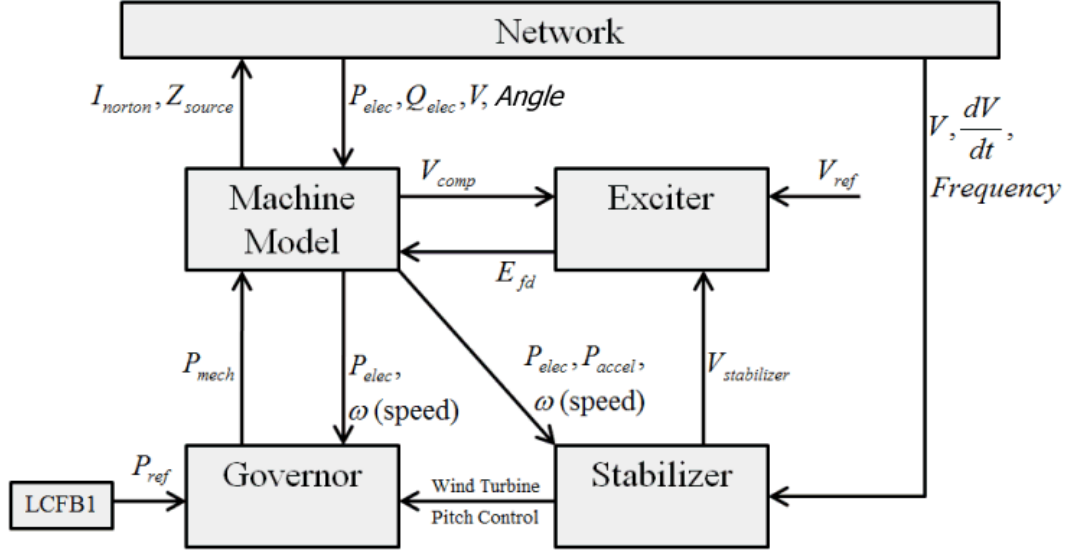


Figure 3.1 Typical interface of machine models with the power network.

The schematic representation of the generator's interface with the power network is shown in Figure 3.1, where the machine can be modeled either as a Thevenin equivalent (voltage source behind an impedance) or a Norton equivalent (current source in parallel with an admittance). Beyond the machine model itself, control loops can be incorporated to ensure stable and reliable generator operation under abnormal grid conditions. These controllers primarily regulate the real and reactive power of the generator. Specifically, the load frequency controller (LCFB1) manages real power and system frequency, while the exciter system governs the generator's voltage magnitude. Oscillations arising during disturbances are mitigated by the power system stabilizer (PSS). The modeling of each of these control blocks, as considered in this study, is detailed in the following sections.

3.2.1 GENCLS Model

The classical generator (GENCLS) model, illustrated in Figure 3.2, represents the most simplified synchronous machine model used in stability studies. It is modeled as a voltage source behind an impedance, where the impedance is assumed to be the transient impedance ($z_{str} = r_{str} + jX_{str}$).

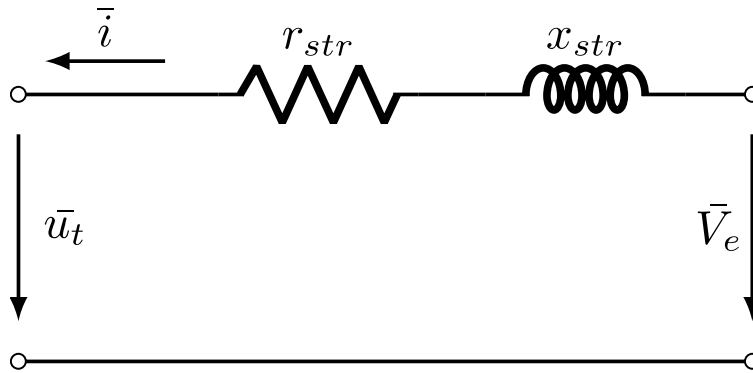


Figure 3.2 Equivalent circuit representation of GENCLS.

This model is typically applied to cylindrical rotor machines. The GENCLS model does not explicitly consider rotor winding flux linkages and assumes a constant mechanical input power P_m (i.e., a blocked governor). Additionally, excitation system dynamics are neglected, allowing the internal voltage magnitude $\bar{V}_e$ to be treated as constant. This model comprises of two states, as described by the equation in (3.1). Owing to its simplifications, it is computationally efficient, though this comes at the expense of reduced accuracy.

3.2.2 Model 2.2

The Model 2.2 version implemented in PowerFactory is referred to as the “*PowerFactory Standard Model*”. In this model, the rotor d -axis is represented by the field winding and a single d -axis damper winding, while the rotor q -axis is modeled with 1 q - and 2 q -axis damper windings. The corresponding equivalent circuit diagrams in the dq domain, represented in the rotor reference frame, are shown in Figure 3.3 and Figure 3.4, respectively. The stator and rotor equations are derived from the equivalent circuit, with parameter notations summarized in Table 3.1. PowerFactory supports simulation in both the RMS and EMT domains; in this work, we present the relevant equations for RMS simulation.

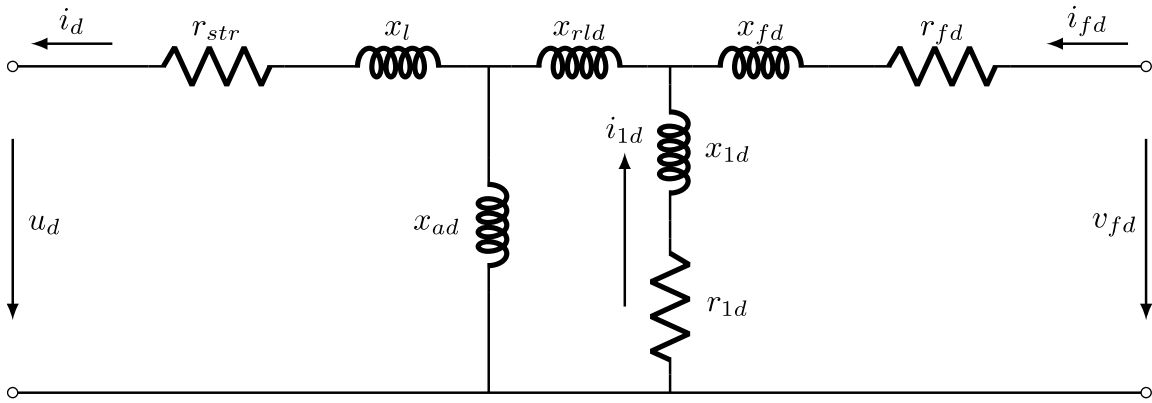


Figure 3.3 Equivalent circuit in d -axis.

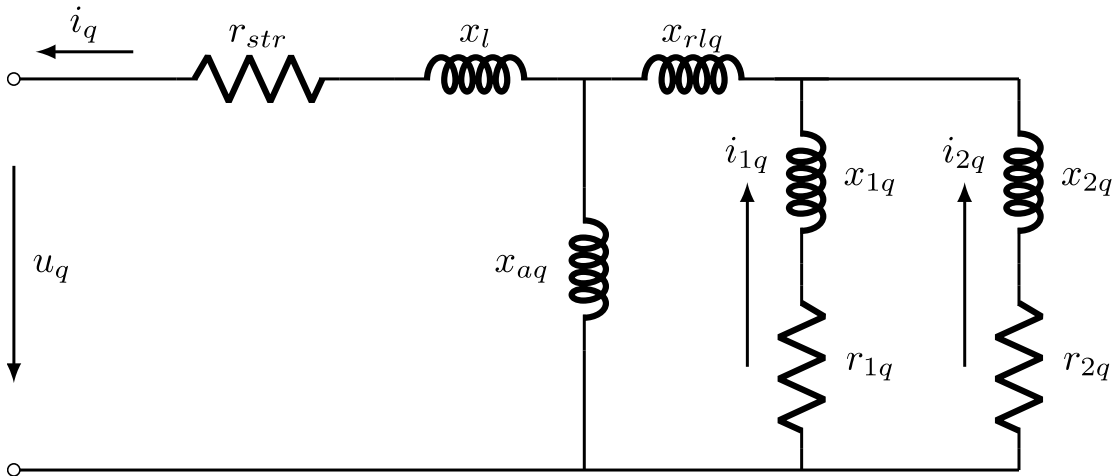


Figure 3.4 Equivalent circuit in q -axis.

Table 3.1 Parameters of the synchronous machine

Symbol	Description
r_{str}	Stator resistance
x_l	Stator leakage reactance
x_{fd}	Reactance of field winding (d -axis)
x_{adq}	Magnetizing reactance in dq -axis
x_{1dq}	Reactance of 1- dq damper winding
x_{2dq}	Reactance of 2- dq damper winding
r_{fd}	Resistance of field winding
r_{1dq}	Resistance of 1- dq damper winding
r_{2q}	Resistance of 2- q damper winding
x_{rld}	Coupling reactance between field and damper winding
x_{rlq}	Coupling reactance between q -axis damper winding

The stator voltages are given by (3.2), where ω is the rotor speed and ω_s is the synchronous speed.

$$\begin{aligned}
 u_d &= -r_{str}i_d - \omega\psi_q + \frac{1}{\omega_s} \frac{d\psi_d}{dt} \\
 u_q &= -r_{str}i_q + \omega\psi_d + \frac{1}{\omega_s} \frac{d\psi_q}{dt} \\
 u_0 &= -r_{str}i_0 + \frac{1}{\omega_s} \frac{d\psi_0}{dt}
 \end{aligned} \tag{3.2}$$

The stator flux equations are expressed as given by (3.3), where ψ_{adq} is the magnetizing flux.

$$\begin{aligned}
 \psi_d &= \psi_{ad} - x_l i_d \\
 \psi_{dq} &= \psi_{aq} - x_l i_q \\
 \psi_{ad} &= x_{ad}(i_{1d} + i_{fd} - i_d) \\
 \psi_{aq} &= x_{aq}(i_{1q} + i_{2q} - i_q)
 \end{aligned} \tag{3.3}$$

On the other hand, the rotor flux linkages in dq -domain are expressed in the following form

$$\begin{aligned}
 \psi_{fd} &= -x_{ad}i_d + (x_{ad} + x_{rld} + x_{fd})i_{fd} + (x_{ad} + x_{rld})i_{1d} \\
 \psi_{1d} &= -x_{ad}i_d + (x_{ad} + x_{rld})i_{fd} + (x_{ad} + x_{rld} + x_{1d})i_{1d} \\
 \psi_{1q} &= -x_{aq}i_q + (x_{ad} + x_{rlq})i_{2q} + (x_{aq} + x_{rlq} + x_{1q})i_{1q} \\
 \psi_{aq} &= -x_{aq}i_q + (x_{ad} + x_{rlq} + x_{2q})i_{2q} + (x_{aq} + x_{rlq})i_{1q}
 \end{aligned} \tag{3.4}$$

By introducing subtransient fluxes (ψ''_{dq}), the stator fluxes can be expressed as

$$\begin{aligned}
 \psi_d &= \psi''_d - x''_d i_d \\
 \psi_q &= \psi''_q - x''_q i_q \\
 \psi''_d &= k_{fd}\psi_{fd} + k_{1d}\psi_{1d} \\
 \psi''_q &= k_{1q}\psi_{1q} + k_{2q}\psi_{2q}
 \end{aligned} \tag{3.5}$$

Equation (3.2) can now be expressed using these equations as

$$\begin{aligned} u_d &= u_d'' - r_{str} i_d + \omega x_q'' i_q - \frac{x_d''}{\omega_s} \frac{di_d}{dt} \\ u_q &= u_q'' - r_{str} i_q - \omega x_q'' i_q - \frac{x_q''}{\omega_s} \frac{di_q}{dt} \end{aligned} \quad (3.6)$$

where u_{dq}'' are given by (3.7), where k_{1dq} , k_{2dq} and k_{fd} are the coupling factors.

$$\begin{aligned} u_d'' &= -\omega(k_{1q}\psi_{1q} + k_{2q}\psi_{2q}) + \left(\frac{k_{fd}}{\omega_s} \frac{d\psi_{fd}}{dt} + \frac{k_{1d}}{\omega_s} \frac{d\psi_{1d}}{dt}\right) \\ u_q'' &= -\omega(k_{fd}\psi_{fd} + k_{1d}\psi_{1d}) + \left(\frac{k_{1q}}{\omega_s} \frac{d\psi_{1q}}{dt} + \frac{k_{2q}}{\omega_s} \frac{d\psi_{2q}}{dt}\right) \end{aligned} \quad (3.7)$$

The rotor voltage equations in the dq -domain are given by (3.8).

$$\begin{aligned} \frac{d\psi_{fd}}{dt} &= -\omega_s r_{fd} i_{fd} + v_{fd} \\ \frac{d\psi_{1d}}{dt} &= -\omega_s r_{1d} i_{1d} \\ \frac{d\psi_{1q}}{dt} &= -\omega_s r_{1q} i_{1q} \\ \frac{d\psi_{2q}}{dt} &= -\omega_s r_{2q} i_{2q} \end{aligned} \quad (3.8)$$

The rotor currents are expressed as given in (3.9)

$$\begin{aligned} i_{fd} &= k_{fd} i_d + \frac{X_{1d}\psi_{fd} - (x_{ad} + x_{rl})\psi_{1d}}{X_{dd}} \\ i_{1d} &= k_{fd} i_d + \frac{X_{fd}\psi_{fd} - (x_{ad} + x_{rl})\psi_{1d}}{X_{dd}} \\ i_{1q} &= k_{1q} i_q + \frac{X_{2q}\psi_{1q} - (x_{aq} + x_{rlq})\psi_{2q}}{X_{qq}} \\ i_{2q} &= k_{2q} i_q + \frac{X_{1q}\psi_{2q} - (x_{aq} + x_{rlq})\psi_{1q}}{X_{qq}} \end{aligned} \quad (3.9)$$

where $X_{dd} = (x_{ad} + x_{rl})x_{1d} + x_{fd}x_{1d}$, $X_{qq} = (x_{aq} + x_{rlq})(x_{1q} + x_{2q}) + x_{2q}x_{1q}$, $X_{fd} = x_{ad} + x_{rl} + x_{fd}$, $X_{1dq} = x_{adq} + x_{rl dq} + x_{1dq}$, and $X_{2q} = x_{aq} + x_{rlq} + x_{2q}$.

The torque equation is expressed as

$$T_{elec} = \psi_d i_q - \psi_q i_d \quad (3.10)$$

The above set of equations, together with (3.1), are solved to determine the state variables of the generator. In the RMS simulation, in addition to positive sequence torque, PowerFactory computes the negative sequence torque using $T_{elec2} = (r_{2sy} - r_{str})i_2^2$, and the net torque is given by $T_{elec} + T_{elec2}$. Consequently, this model evaluates six states - ψ_{fd} , ψ_{1d} , ψ_{1q} , ψ_{2q} , ω , and δ , making it more detailed and reliable compared to GENCLS model.

3.2.3 Governor Model

Under synchronous operation, electrical power and mechanical power are balanced. However, during grid disturbances, electrical power demand can exceed the available mechanical power. This imbalance is initially compensated by the kinetic energy stored in the rotating mass of the generator. As the kinetic energy decreases, the generator speed and frequency decline. The governor (primary frequency response) senses this speed deviation and adjusts the valve input, thereby modifying the mechanical power until a new steady-state operating point is established. Among several governor models in the literature, the simplest IEEEG1 (Figure 3.5 [65]) is considered here for the study.

This model enables the control of cross-compound units and incorporates the main valve rate limits (U_o and U_c). The droop characteristic is represented by the constant K , while the lead-lag dynamics are captured through the time constants T_1 and T_2 . A speed-deviation dead band is defined by DB. The odd-indexed gains (K_1 – K_7) and time constants (T_1 – T_7) represent the high-pressure turbine, whereas the even-indexed parameters characterize the low-pressure turbine. The reference power signal (P_{ref}) is provided by the load frequency controller.

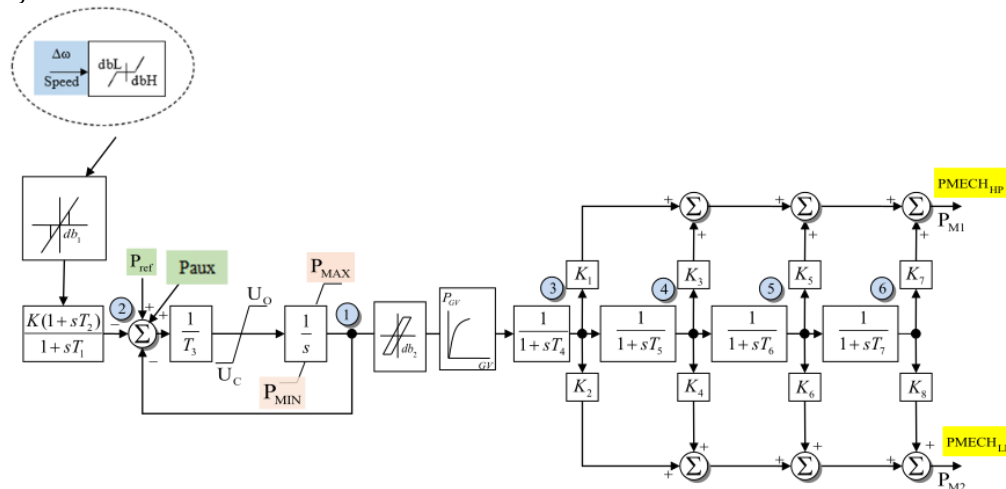


Figure 3.5 Mathematical representation of the IEEEG1 turbine governor model.

3.2.4 Exciter Model

Generator terminal voltage and reactive power flow are regulated through excitation control, implemented using an automatic voltage regulator (AVR). Variations in reactive power result in corresponding changes in terminal voltage magnitude, which are detected and converted into a DC signal. The difference between this measured signal and the reference set point forms the error signal, which is amplified and applied to the field winding. This process adjusts the internal generated EMF, thereby altering the reactive power output. The system settles at a new equilibrium point such that the terminal voltage matches the desired reference value. Excitation systems are generally classified into DC, AC, and static types. In static exciters, the excitation power is supplied through transformers and rectifiers. The terminal voltage is processed through controlled or uncontrolled rectifiers, which regulate the output to deliver the required current to the generator field winding.

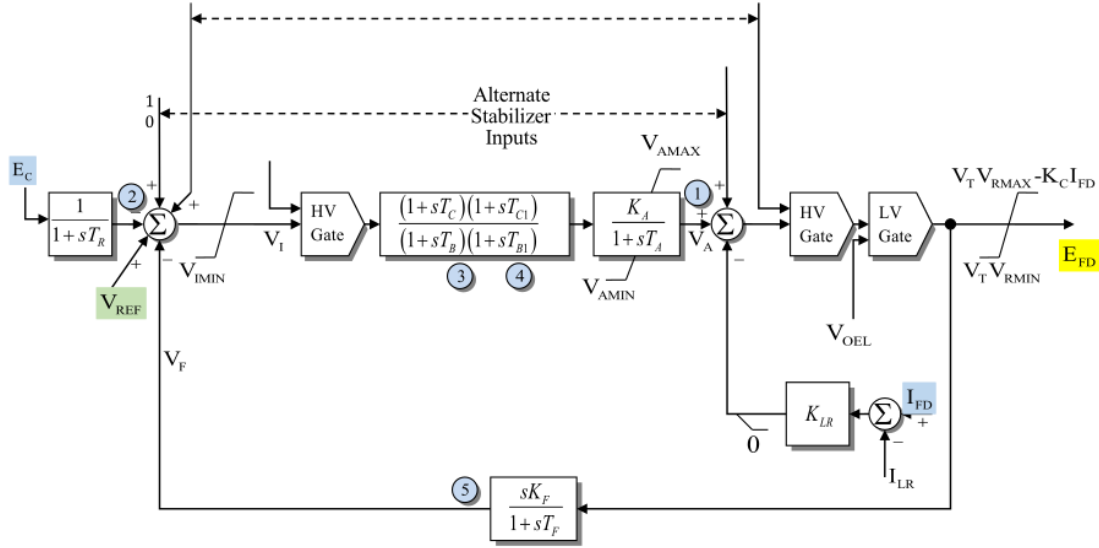


Figure 3.6 Type ST1A exciter model.

In this study, emphasis is placed on the fast-acting static Type ST1A excitation system, as illustrated in Figure 3.6 [65]. The field winding voltage is controlled using a thyristor bridge rectifier, where regulation is achieved through adjustment of the firing angle. The desired stator terminal voltage is specified by the reference signal V_{ref} , while an auxiliary input V_{sin} from the PSS is incorporated to enhance damping of electromechanical oscillations. The voltage regulator is characterized by a gain K_A and time constant T_A . Excitation system stabilization is provided through a feedback loop, modeled as a first-order transfer function with gain K_F and time constant T_F . To ensure protection of the generator rotor and exciter, a field current limiter (I_{LR}) is included in the system. Owing to the inherently small-time constants of the exciter, external stabilization is typically unnecessary; however, transient gain reduction or enhancement may be required. This is accomplished through appropriate tuning of parameters T_B , T_C , T_{B1} , and T_{C1} .

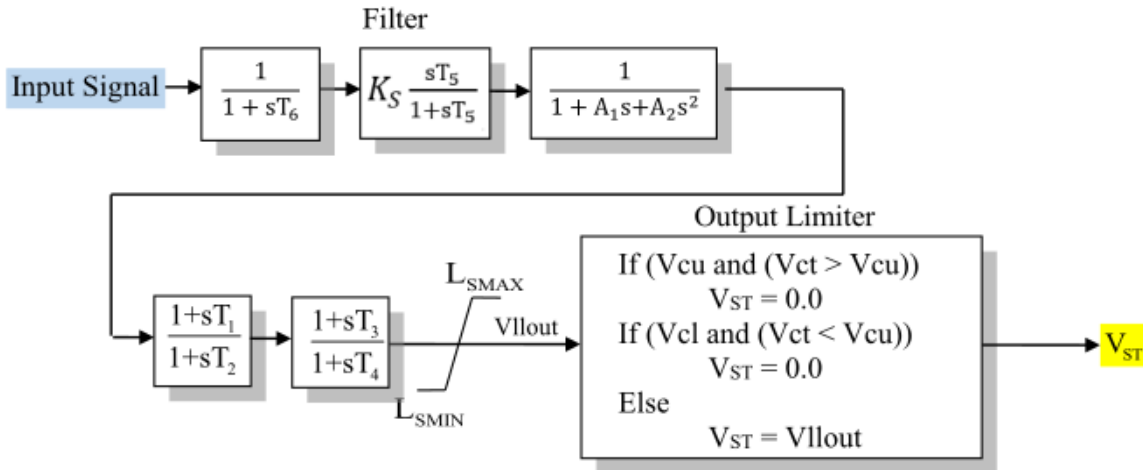


Figure 3.7 General schematic of power system stabilizer (PSS1A).

3.2.5 Power System Stabilizer (PSS)

The Power System Stabilizer (PSS) is employed to mitigate power system oscillations by modulating the excitation system, thereby producing an additional damping torque that is in phase with rotor oscillations. In this study, the PSS1A stabilizer model, illustrated in Figure 3.7 [65], is adopted. The input signal to the PSS may be rotor speed, rotor speed deviation, frequency, or electrical power output. The stabilizer gain is denoted by K_S , while the washout block is characterized by a time constant T_5 . Appropriate tuning of parameters A_1 and A_2 enables the desired trade-off between low-frequency response and attenuation of high-frequency torsional modes. Lead-lag compensation is provided by time constants T_1 - T_4 , ensuring effective phase compensation for damping electromechanical oscillations.

Up to this point, the machine modeling and associated controllers have been described. It is important to note that, despite the large number of state variables involved, the generator rotor angle (δ) and speed (ω) are sufficient to determine the CCT for a given fault scenario. These two variables directly capture the rotor dynamics that govern system stability. Consequently, the Koopman operator-based models in this study employ only δ and ω as input features, ensuring both computational efficiency and physical interpretability.

3.3 IBR Modeling for BPS Studies

The IBRs integration is rapidly increasing, making accurate modeling essential for BPS stability studies. IBRs include photovoltaic (PV) systems, energy storage systems, and wind power systems, among others. These resources are typically connected through grid converters (inverters) controlled by pulse width modulation (PWM) techniques. The converter outputs are then passed through filters (L , LC , or LCL) to meet harmonic performance requirements. Consequently, the overall impact of IBRs on the power network can only be captured through detailed EMT simulations, which require solving the differential equations associated with IBRs.

Many IBRs are typically connected in parallel and interfaced with the grid through transformers at the point of interconnection (POI). The current output from each IBR is delivered to the POI through a collector cable system. Accurate assessment of such systems requires detailed modeling, which involves solving numerous differential equations. Moreover, since IBRs are expected to provide grid support under adverse operating conditions, their control systems must also be explicitly represented within simulation platforms.

To address the resulting computational complexity, aggregated modeling approaches are widely adopted for stability studies. In these models, individual IBR units (e.g., PV or wind systems) are represented collectively as a single equivalent IBR, while inverter dynamics are captured using average inverter models. This simplification significantly reduces computation time in IBR-rich systems and enables representation of plant-level controllers as a single or a limited number of subsystems. Several notable studies on IBR aggregation have been reported in the literature. In this work, a simplified aggregated PV model is employed. Additionally, the DC-side dynamics of the PV system are neglected, with the PV arrays represented as a constant DC voltage source.

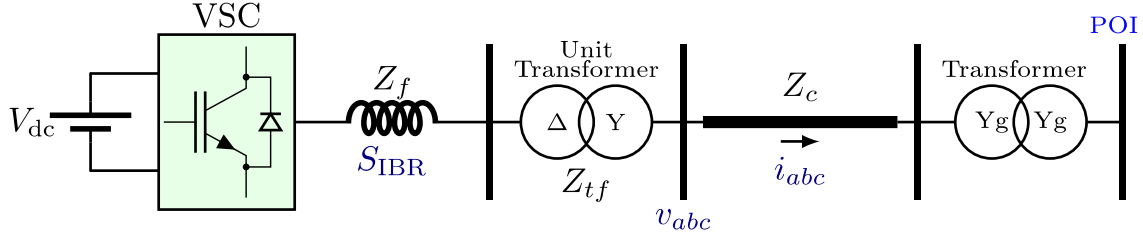


Figure 3.8 Typical IBR plant layout.

The typical interface of an IBR plant to the grid at the POI is illustrated in Figure 3.8. On the DC side, the dynamics are simplified and represented as a constant voltage source V_{dc} . The voltage source converter (VSC) converts the DC power from the PV plant into AC, which is delivered to the POI through filters and cables. Inverters regulate active and reactive power injections into the grid and provide essential grid support services.

On the AC side, the inverter output is stepped up to medium-voltage levels via pad-mounted (unit) transformers. The power from multiple IBR units is then aggregated through the collector system (Z_c), consisting of overhead or underground cables, before being stepped up again to the POI voltage level for interconnection with the wider network. To enable power control and monitoring, sensors are placed at the unit transformers and along the collector system. The filter parameters (Z_f) are designed to satisfy total harmonic distortion (THD) requirements. In this study, a simple L -filter is adopted, as it is sufficient for high-power applications.

3.3.1 Grid-following inverter control

The grid-following (GFL) converter operates as a controlled current source. It tracks the grid phase angle and synchronizes with the grid using a phase-locked loop (PLL). Desired active and reactive power injections are achieved by regulating the corresponding current components. A high-level representation of the GFL strategy is shown in Figure 3.9 [66].

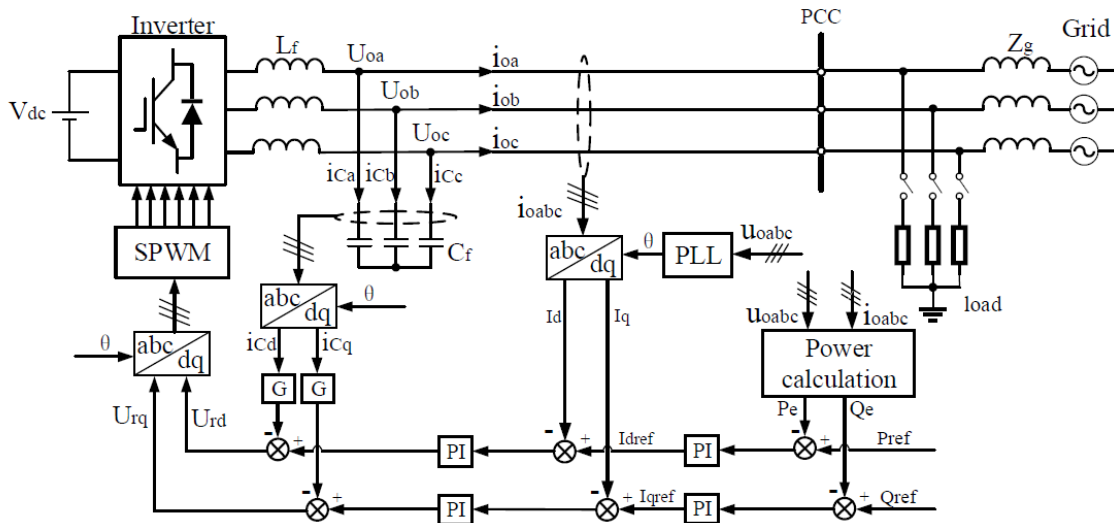


Figure 3.9 PQ control strategy of GFL.

The phase angle of the POI (also referred to as point of common coupling (PCC)) is tracked using a phase-locked Loop (PLL). A dual vector control strategy is applied to regulate the active and reactive current injections into the grid. The control is implemented in the dq reference frame, with the transformation defined by (3.11), where ω_n denotes the reference frame frequency derived from the PLL.

$$T_{dq}(\omega_n) = \begin{bmatrix} \cos(\omega_n t) & \sin(\omega_n t) \\ -\sin(\omega_n t) & \cos(\omega_n t) \end{bmatrix} \quad (3.11)$$

Based on this choice of orientation, the powers are given by (3.12).

$$P = 1.5 [v_d i_d + v_q i_q] \quad \text{and} \quad Q = 1.5 [v_q i_d - v_d i_q] \quad (3.12)$$

Under balanced operation ($v_d = \hat{V}$), the current references (i_{dq}^*) are provided by (3.13).

$$\begin{aligned} i_d^* &= \frac{2}{3} \frac{P^*}{v_d} \\ i_q^* &= -\frac{2}{3} \frac{Q^*}{v_q} \end{aligned} \quad (3.13)$$

i_{dq}^* are processed through PI controllers to generate the corresponding reference voltages which are used to control the inverter using PWM techniques.

Under unbalanced conditions, along with positive sequence components, negative sequence components come into picture. In such scenario, the current references are generated using (3.14)

$$\begin{aligned} i_{dq1}^* &= (v_{d1} + jv_{q1}) \left(\frac{P^*}{A} - \frac{Q^*}{B} \right) \\ i_{dq2}^* &= -k_\phi (v_{d2} + jv_{q2}) \left(\frac{P^*}{A} - \frac{Q^*}{B} \right) \end{aligned} \quad (3.14)$$

where the subscript 2 indicates negative sequence, k_ϕ is the unbalance compensation factor, $A = v_{d1}^2 + v_{q1}^2 - k_\phi (v_{d2}^2 + v_{q2}^2)$ and $B = v_{d1}^2 + v_{q1}^2 + k_\phi (v_{d2}^2 + v_{q2}^2)$. The value of k_ϕ dictates the level of unbalanced compensation provided by the IBR. For obtaining the negative sequence components, the three phase quantities are passed through $T_{dq}(-\omega_n)$.

The detailed IBR control scheme adopted for this study is illustrated in Figure 3.10, where u_{gdq12} represent the point of measurement (POM) sequence voltages in the dq domain. The POM voltages, denoted as v_{abc} , are shown in Figure 3.8. The IBRs regulate reactive power based on the voltage magnitude measured at the POM: when the voltage drops below 0.9 p.u., reactive power is injected, whereas for voltages above 1.05 p.u., excess VARs are absorbed. This voltage condition is detected by the disturbance block, which activates the 'lv_on' and 'hv_on' signals accordingly. The corresponding current references during such disturbances are generated in compliance with IEEE P2800-2022. The collector current measured at the POM (i_{dq}^{SRF}) is then compared against the reference currents, and the resulting error is processed through a PI controller to generate the gating signals for the inverter.

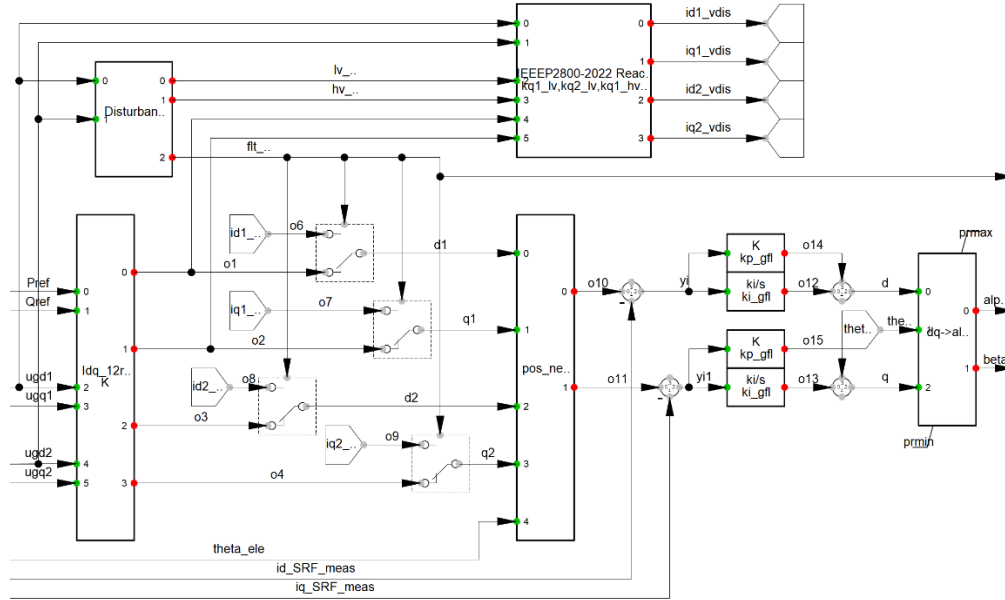


Figure 3.10 Detailed IBR control in PowerFactory.

As shown in Figure 3.10, P_{ref} and Q_{ref} serve as inputs for current reference generation, with Q_{ref} determined by the voltage magnitude at the POM. By formulating reference signals based on voltage conditions and incorporating current injection limits, the control inherently safeguards the IBRs against potential damage.

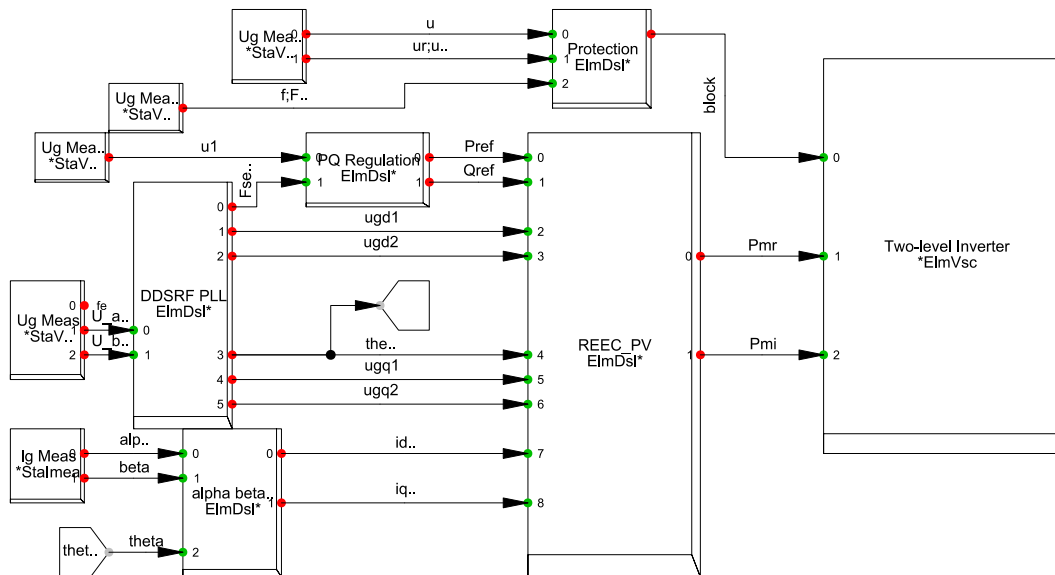


Figure 3.11 Schematic of VSC control in PowerFactory.

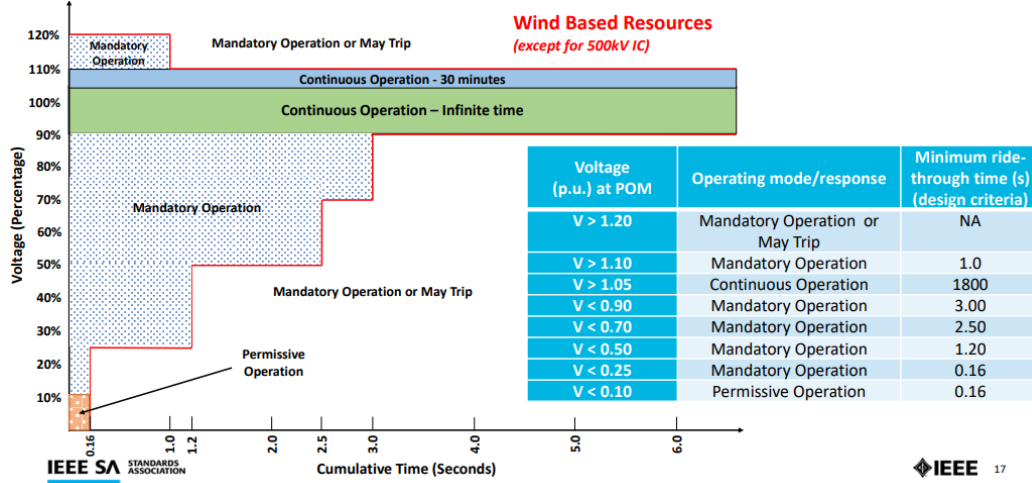


Figure 3.12 Voltage ride-through chart.

Table 3.2 VRT settings considered for the study

V [p.u.]	Ride through time [s]
> 1.2	0.086
< 0.8	3
< 0.25	0.16

The complete VSC control with the control implemented in PowerFactory is shown in Figure 3.11. The PLL is used to generate the phase angle of the system. REEC_PV is the current controller of the IBR module. The PowerFactory measurements provide the signals in stationary reference frame. The reference powers are generated using PQ regulation block. The IBR is equipped with voltage ride-through (VRT) settings (illustrated in Figure 3.12 [68]), mimicking real-world scenarios and settings considered for the analysis are shown in Table 3.2. Depending on the POM voltage and the ride-through time, the protection logic triggers a trip signal to the VSC converter. For this study, complete cessation is assumed, meaning the IBR will not be reconnected even after the system restores to the normal operating range.

Another critical component of the IBR plant is the PLL, which significantly influences the plant's dynamics. For this study, the PLL considered is the Double-Decoupled Synchronous Reference Frame (DDSRF-) PLL illustrated in Figure 3.13. This PLL utilizes two dq frames, with their outputs fed back to form a closed-loop operation, ensuring accurate synchronization with the grid while maintaining stability and good dynamic response. The dq quantities are passed through a low-pass filter with a time constant of 0.01 s, and the frequency is estimated using the q-axis component. The detailed analysis of DDSRF-PLL is presented in [67]. The general schematic of PLL is shown in Figure 3.14, where the filtered dq quantities are used to estimate the grid frequency ($\hat{\omega}$). Estimated phase is fed-back to dq frame forming a closed-loop action, and thus the desired transient performance of PLL greatly depends on the choice of PI parameters.

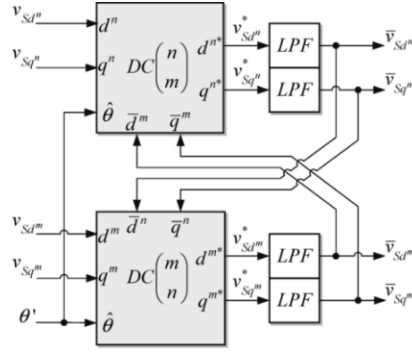


Figure 3.13 Decoupling network to obtain the sequence components.

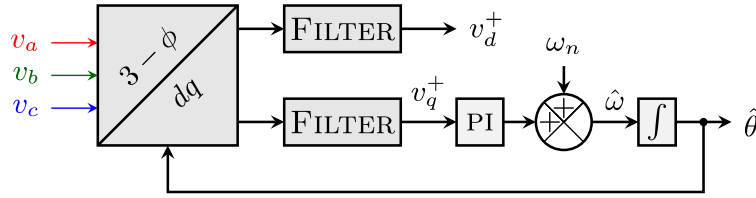


Figure 3.14 General schematic of PLL.

The governing equation of PLL is given by

$$G(s) = V^+ \left(\frac{1}{\tau s + 1} \right) \left(\frac{s k_p + k_i}{s} \right) \left(\frac{1}{s} \right) \quad (3.15)$$

where τ is the low-pass filter time constant, and k_p and k_i are the PI controller parameters. Thus the performance of the PLL greatly depends on the PI parameters and is also sensitive to input voltage fluctuations. For the analysis, $k_p = 50$ and $k_i = 0.3$ are chosen for the study. The corresponding bode plot is shown in Figure 3.15. The designed PLL has a phase-margin of 65.5° , resulting in a good transient response. Furthermore, the high open-loop DC gain satisfies the steady-state error criterion.

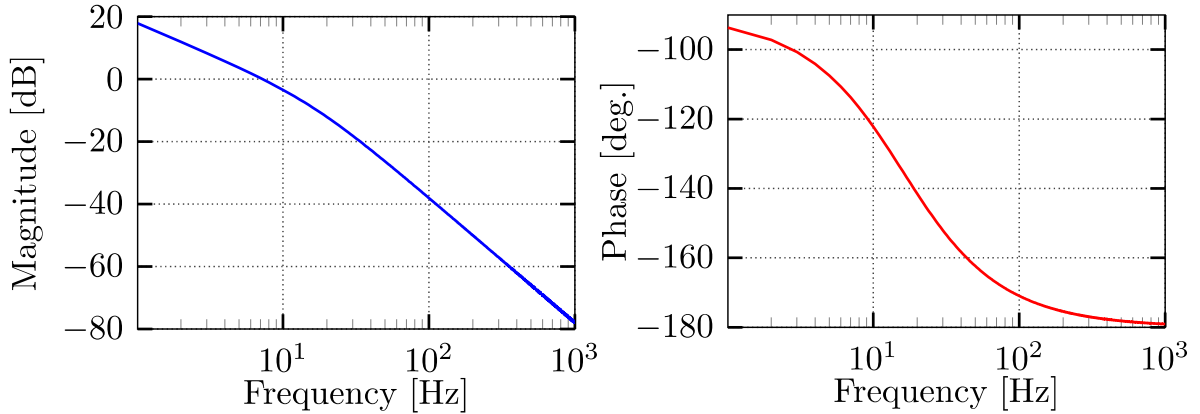


Figure 3.15 Bode response of PLL considered for the study.

3.4 Hybrid EMT+Phasor models

The traditional grid dynamics are captured using phasor domain simulation which is suitable for large-system studies and cannot capture the fast-acting transients. On the other hand, the EMT simulations capture the fast-acting transients and requires one to solve time-domain equations, which can be computationally exhaustive when considered for modern power grids. Thus to have a reasonable trade-off, hybrid RMS-EMT power system models were developed that combines one domain with another domain. In such models, IBRs and other components where fast dynamics are critical are represented in the EMT domain, while the remainder of the grid is modeled in the RMS domain (either balanced or unbalanced), ensuring both fidelity and scalability in simulation.

Hybrid models can be viewed as a specialized form of co-simulation, in which one simulation domain is interfaced with another, either externally or internally. In this work, DIgSILENT PowerFactory is employed to conduct the co-simulation studies. The schematic representation of the co-simulation domains is shown in Figure 3.16, where different regions of the network are represented in either the RMS or EMT domain. These domains are separated by co-simulation boundaries (indicated as interface lines), which delineate the components belonging to each region. PowerFactory provides two approaches for defining such boundaries: branch-based and node-based splitting. In this study, the branch-based method is adopted, with the co-simulation regions defined across the collector cable connecting the POI transformer and the IBR transformer.

Once the boundaries are defined, the next step is to initialize the system under study. For co-simulation, a multi-domain initialization is performed, where the sampling times of the RMS and EMT regions are specified. In addition, a synchronization period is defined, which determines the rate at which data is exchanged across the co-simulation interface. The step sizes of the RMS and EMT solvers can be set independently. In this study, an EMT step size of $100\ \mu\text{s}$, an RMS step size of $500\ \mu\text{s}$, and a synchronization period equal to the RMS step size are considered. Adjusting these time parameters can accelerate or slow the simulation but does not affect the underlying system stability. The mechanisms of data transfer between RMS and EMT domains differ in each direction. While this process occurs internally within the simulation environment, detailed descriptions cannot be provided here due to tool-specific restrictions.

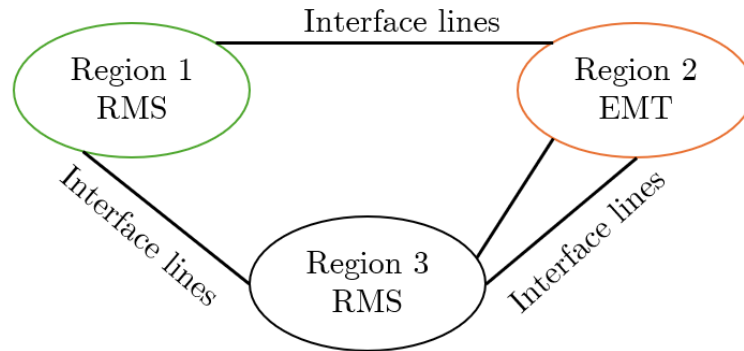


Figure 3.16 Schematic of co-simulation domains.

3.5 Data generation for study

In PowerFactory, the data required for analysis is generated using a Python script that employs the PowerFactory Python API to automate the simulation process. The workflow consists of system initialization, creation and simulation of fault events, storage of the simulation results, and subsequent removal of the events. In this study, only bus faults are considered. Faults are introduced at all buses with varying fault impedances and fault clearing times, and the procedure is repeated in a loop until all scenarios are evaluated. For each scenario, the relevant variables of interest are exported to a “.csv” file, which is subsequently used for the data-driven analysis. The complete procedure is summarized in Algorithm 2, which is formulated in a generalized manner and can be applied with any black-box simulator.

Algorithm 2 Script automation for fault study using PowerFactory.

```
1: Initialize the *.pfd file in PowerFactory using API
2: Access the bus names and generate the fault impedances ( $Z_f$ ) and clearing times ( $t_c$ ) of interest for study
3: Select the variables to monitor, simulation end time ( $t_{end}$ ), type of simulation (RMS or EMT or co-simulation)
4: Generate the output folder path to save the results
5: for  $t_c \leftarrow \text{max to min}$  do
6:   for  $Z_f \leftarrow \text{max to min}$  do
7:     for all the bus names do
8:       Create the fault event using bus names,  $Z_f$  and  $t_c$ 
9:       Run the power-flow and initialize for dynamic studies
9:       Run the simulation for time duration of  $t_{end}$ 
10:      Write the results to the output folder
11:      Delete the fault event, results and reset the calculation
12:    end for
13:  end for
14: end for
```

In this study, different fault impedances and clearing times are evaluated using nested for loops. A fault event is created through the Python API, which requires specification of the fault location, type (balanced or unbalanced), fault duration, and the clearing instant (either zero or immediate). Once the event is defined, the co-simulation is initialized and executed for the desired simulation duration. After completion, the corresponding results are exported to the designated folder. PowerFactory maintains separate result folders for RMS and EMT simulations, which requires careful handling during post-processing. Following data storage, the fault event and associated results are cleared. It is also noted that initialization with user-defined models requires additional time to reach steady state, and PowerFactory does not support snapshot initialization in co-simulation mode. Consequently, the system must be re-initialized after each fault event.

With the selected simulation settings, each case requires approximately 1 minute and 30 seconds, resulting in a total simulation time of about 25 hours for 1000 cases. In addition, as the number of

recorded variables increases, the corresponding file size also grows, which introduces additional complexity in data access and analysis.

In principle, the number of possible simulations is infinite, since fault conditions can vary continuously in type, location, impedance, and clearing time. Therefore, representative sampling is essential to make the problem tractable. The dataset was constructed to span a wide variety of operating conditions by varying fault locations, fault impedances, and clearing times, with randomized sampling ensuring continuous coverage of the time axis.

This design deliberately includes both stable and unstable trajectories, allowing the Koopman operator to learn the critical boundary between secure and insecure system responses. Clearing times were distributed on a logarithmic scale to provide denser sampling in the region where system dynamics are most sensitive, while still capturing longer-duration cases. As a result of this distribution, the dataset may naturally contain a higher proportion of stable scenarios compared to unstable ones, reflecting the fact that shorter clearing times are more likely to maintain stability. The overall approach balances representativeness with computational feasibility, ensuring that the dataset is rich enough to train the operator without introducing unnecessary redundancy. As a result, the generated dataset captures a broad spectrum of system behaviors while remaining tractable for large-scale analysis.

The generator states, rotor speed (ω) and rotor angle (δ), are the primary variables of interest and are stored in .csv files. For clarity and ease of access, the files are named according to the convention “Bus No- t_c -Zfi-x.csv”, where t_c denotes the fault clearing time, Zfi represents the i^{th} fault impedance, and x indicates the fault type (three-phase, two-phase, or single-phase). This naming scheme allows direct identification of the clearing time and fault characteristics from the file name, thereby facilitating analysis of the corresponding system state behavior. For stable cases, the system trajectories remain within the ROA and converge to the pre-fault equilibrium ($\bar{\varphi}(t) = 1$). In contrast, when the system becomes unstable ($\bar{\varphi}(t) = 0$), the trajectories diverge. The primary challenge lies in enabling the algorithm to correctly distinguish between stable and unstable scenarios.

Prior to the data-driven analysis, we require to pre-process the data which typically involves assigning labels 0 (unstable) and 1 (stable) to each scenario. To do this, we use time-averaging function given by

$$f = \min \left(k \times \frac{1}{n} \times \sum_{i=0}^n e^{-\mu(x_i)/\sigma}, 1 \right) \quad (3.16)$$

Where k is the scaling factor, n is the number of observations in the series, $\mu(x_i)$ is the mean of the i^{th} observation, and σ is a threshold parameter that dictates the sensitivity of the exponential decay and, effectively, the spread of the rotor angle distribution. The function is bounded above by 1, ensuring that the output remains normalized. The exponential term provides a decaying weight that decreases as the mean value increases, thereby emphasizing the deviation of system trajectories from the stable operating point.

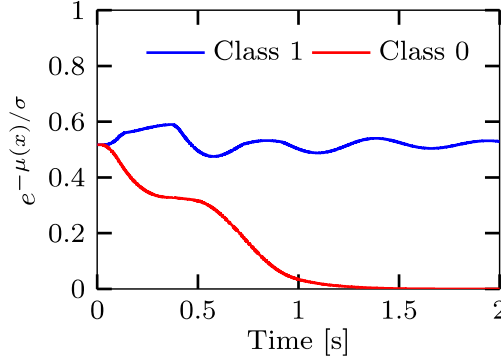


Figure 3.17 Variation of the exponential term for stable and unstable rotor angle trajectories.

In this application, the rotor angles are used as indicators of instability, since under unstable conditions they diverge rapidly to higher magnitudes. To improve robustness, the rotor angles δ are unwrapped before being processed through the time-averaging function. For this study, we select $k = 2$ and $\sigma = 30^\circ$, which were empirically chosen to balance sensitivity and robustness in classification. The variation of exponential term is illustrated in Figure 3.17.

For a stable fault scenario, the mean of the observations remains nearly constant, and therefore the exponential function also stays constant. Such cases are assigned the label 1 (Class 1). In contrast, for unstable trajectories, the mean of the observations increases over time due to the diverging rotor angle, which drives the exponential term to decay toward zero. These cases are labeled as 0 (Class 0). In this way, stable and unstable scenarios are distinguished without explicit knowledge of the fault clearing times. By applying this time-averaging function, each scenario can be systematically classified as stable, when trajectories remain bounded and converge to the pre-fault equilibrium, or unstable, when trajectories diverge. This approach integrates both the dynamic behavior of system states and the statistical distribution of observations into the classification framework.

The complete preprocessing workflow is shown in Figure 3.18, where the full set of simulation data is utilized to generate the training dataset. In addition to the generator states δ and ω , t_c is extracted directly from the file names. Labels are generated based on the δ , and the corresponding system states are stored only at the fault clearing instant. Consequently, the dataset does not include all subsequent stable or unstable trajectories beyond t_c . After all simulation files have been processed, the accumulated states and labels are stored in a '.npy' file. This file serves as the training dataset for the KNN models, which are later used to predict the CCT for new scenarios. It is to be noted that the training dataset is kept separate from the testing data to ensure unbiased evaluation of the model performance.

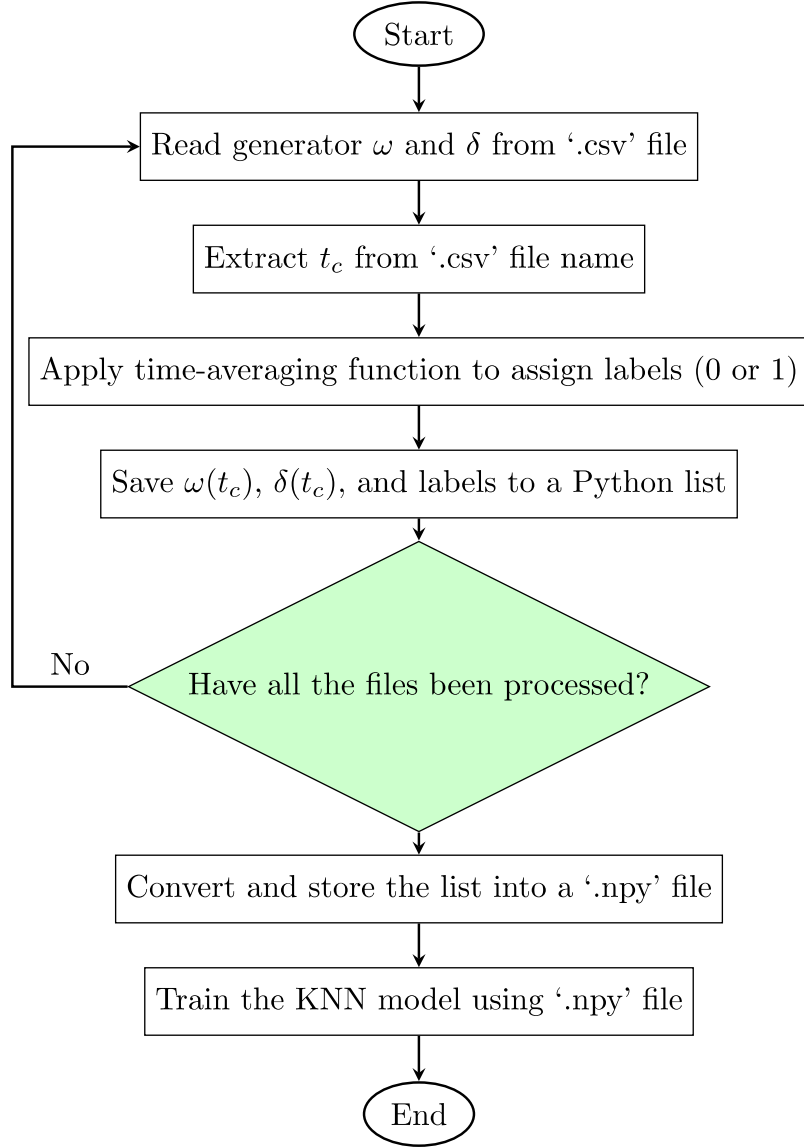


Figure 3.18 Flowchart of the automated simulation and data processing workflow.

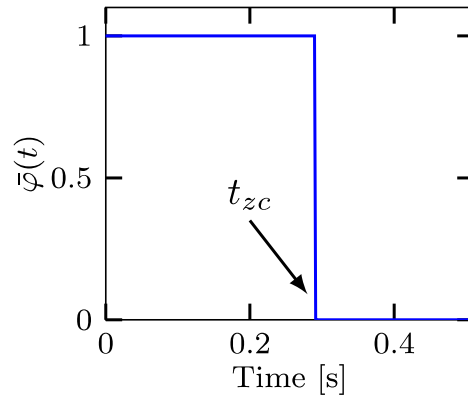


Figure 3.19 Expected response of the indicator function.

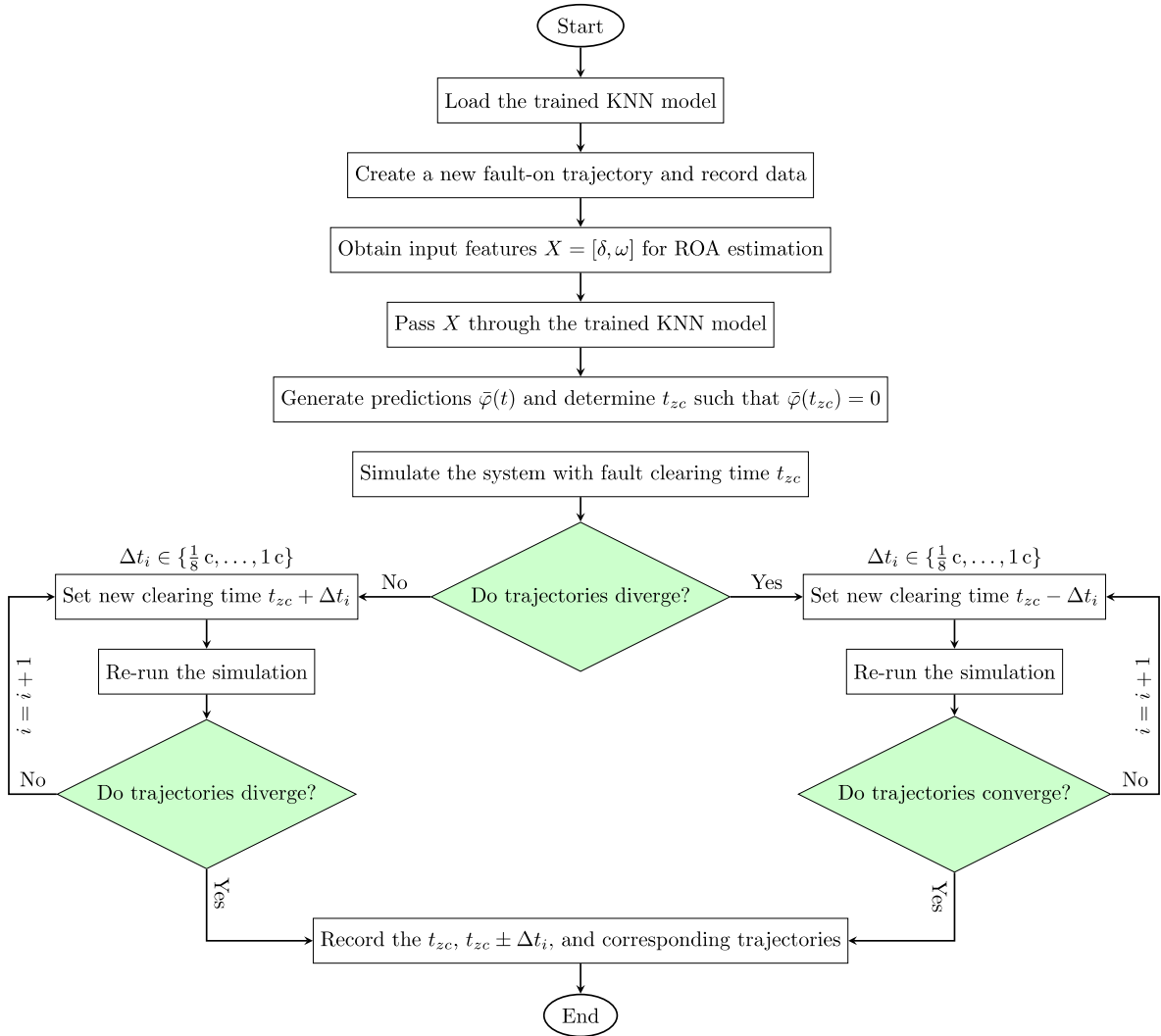


Figure 3.20 Flowchart illustrating the procedure for validating CCT.

The CCT is estimated from the new fault-on trajectory by identifying the instant when the indicator function touches the zero point, denoted as t_{zc} (as illustrated in Figure 3.19). The data-driven model provides predictions along the fault-on trajectories, with the input feature set defined as $X = [\delta, \omega]$. The complete procedure for obtaining the CCT is shown in Figure 3.20, where the model predictions are evaluated over the simulation horizon and the corresponding zero-hitting instant t_{zc} is determined. A new set of simulations is performed by setting t_{zc} as the fault clearing time. Since the estimated zero-hitting instant may or may not correspond exactly to the true CCT of the system, repetitive time-domain simulations are conducted based on the rotor angle trajectories, as follows:

1. If the trajectories converge for a fault clearing time of t_{zc} , additional simulations are carried out with the clearing time set to $t_{zc} + \Delta t$. This process is repeated until the trajectories diverge, and the corresponding fault clearing time at which divergence occurs is recorded along with δ and ω .

2. If the trajectories diverge for a fault clearing time of t_{zc} , additional simulations are carried out with the clearing time set to $t_{zc} - \Delta t$. This process is repeated until the trajectories converge, and the corresponding fault clearing time at which convergence occurs is recorded along with δ and ω .

Here, Δt is varied from $1/8$ c to 1 c, where c indicates cycle (1 c = 16.667 ms). This procedure ensures that the estimated CCT for the given fault lies within the neighborhood of the true CCT.

4. Results and Discussion

In this section, we present numerical results for data-driven transient stability assessment on four power system examples: (1) a two-state single-machine infinite-bus example simulated in Python as an ordinary differential equation (ODE); (2) a four-state two-machine infinite-bus example simulated in Python as an ODE; (3) the ten-machine New-England system simulated with constant power loads in PSSE and in PowerFactory with IBRs; and (4) 240 WECC system in PSSE. Both the kernel method and the time-averaged method were implemented in Python to estimate the CCT of fault scenario. The objective of this section is to demonstrate the effectiveness of the RKHS-based Koopman approximation, along with the time-averaged estimation of the unit eigenfunction, for transient stability analysis. The results further highlight the capability of the proposed data-driven framework in capturing stability characteristics of IBR-rich grids.

4.1 The Single-Machine Infinite-Bus (SMIB) System

Consider a per unit (p.u.) system whose one-line diagram is given in Figure 4.1. Its electromechanical dynamics is described by

$$\dot{\delta} = \omega, \dot{\omega} = -D\omega + P_m - P_e \sin(\delta) \quad (4.1)$$

with damping constant $D = 1.3 \text{ rad}^{-1}\text{s}$, mechanical power input $P_m = 5 \text{ p.u.}$, and electrical power coefficient $P_e = 10 \text{ p.u.}$ which corresponds to the maximum electrical power that can be transferred from this generator. The stable equilibrium point of this system is $[0.53 \text{ rad}, 0]$. Thus, the system under consideration is measure-preserving with respect to the Dirac delta measure at the equilibrium point. We consider the case where a transmission line fault happens and subsequently, P_e deviates from its value under normal operating conditions. In the most extreme case, a bolted fault corresponds to $P_e = 0$. While the value of P_e is determined by the fault location and impedance, for this experiment, we choose a set of values of P_e . The smaller the value of P_e , the lesser the power deliverable to the infinite bus, indicating a more severe fault condition. In this study, we study three fault-on scenarios with $P_e \in \{0.95, 0.75, 0.45\} \text{ p.u.}$

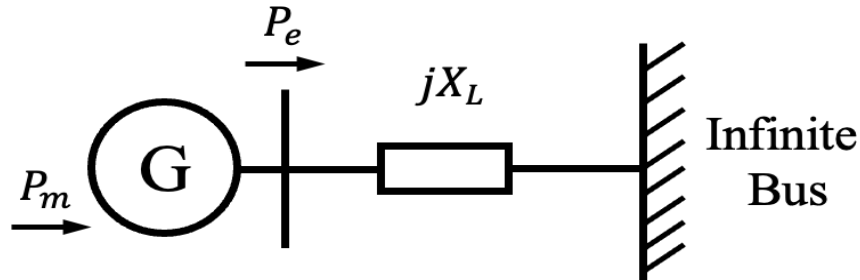


Figure 4.1 SMIB system connected to an infinite bus through a transmission line.

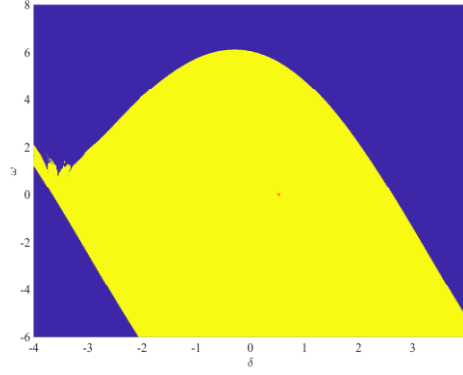


Figure 4.2 A plot of the estimated indicator function which reveals the ROA.

To construct the indicator function using RKHS, we sampled 1500 initial points $x = [\delta, \omega]$ that are uniformly distributed over $[\delta, \omega] \in [-4, 4] \times [-6, 6]$. We then numerically integrated the ODE starting from each x for a time interval of 0.1 s to obtain $x +$. For this 2-D system, we first scaled ω by $\frac{1}{2}$ and then used a Gaussian kernel $\kappa(x_1, x_2) = \exp(-\|x_1 - x_2\|_2^2 / 2 \times 0.5^2)$. The indicator function was then derived from φ_1 (highlighted in Chapter 2) with $\epsilon = 8 \times 10^{-4}$. For the time-averaged indicator function, the same 1500 initial points were simulated for 20s with a time-step of 0.05 s. The seeding function was the radial basis function (RBF) centered at [0.53 rad, 0] with a standard deviation of 0.25 along δ and a standard deviation of 0.5 along ω . $\bar{\varphi}$ was then obtained by the nearest neighbor algorithm in scikit-learn [42]. The resulting $\bar{\varphi}$ is plotted in Figure 4.2.

To compute the CCTs, the indicator function $\bar{\varphi}$ is evaluated along each fault-on trajectory for $P_e \in \{0.95, 0.75, 0.45\}$, and the results are plotted in Figure 4.3. As shown in the figure, the value of $\bar{\varphi}$ remains unity until the fault-on trajectory passes through the boundary of the ROA and it drops to zero. As indicated in Table 4.1, the CCTs from the time-averaging technique matched those obtained via the RKHS, and they both matched the values we obtained from time-domain simulations of the post-fault trajectories starting from various points of the fault-on trajectories.

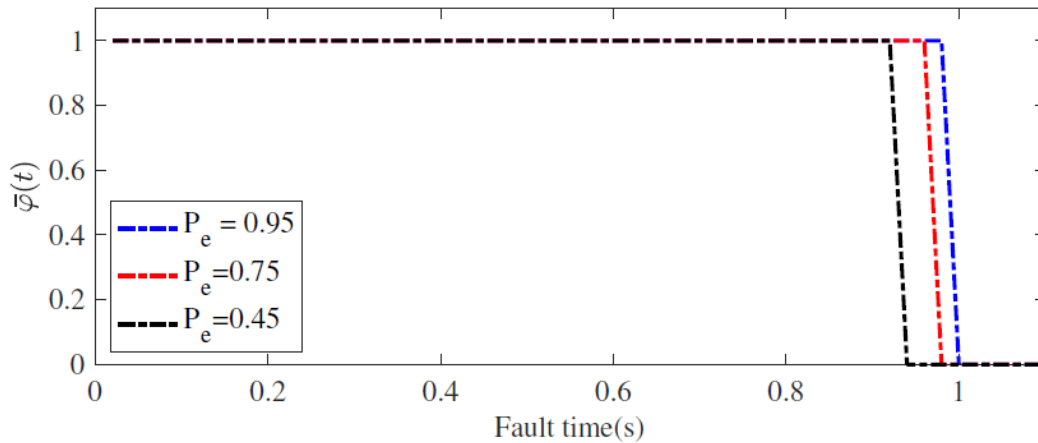


Figure 4.3 Value of the indicator function evaluated at fault-on trajectories with choices of P_e .

Table 4.1 CCT [s] obtained using various for SMIB system

P_e [p.u.]	Time-averaging	RKHS	True Value
0.95	0.98	0.98	0.98
0.75	0.96	0.96	0.96
0.45	0.92	0.92	0.92

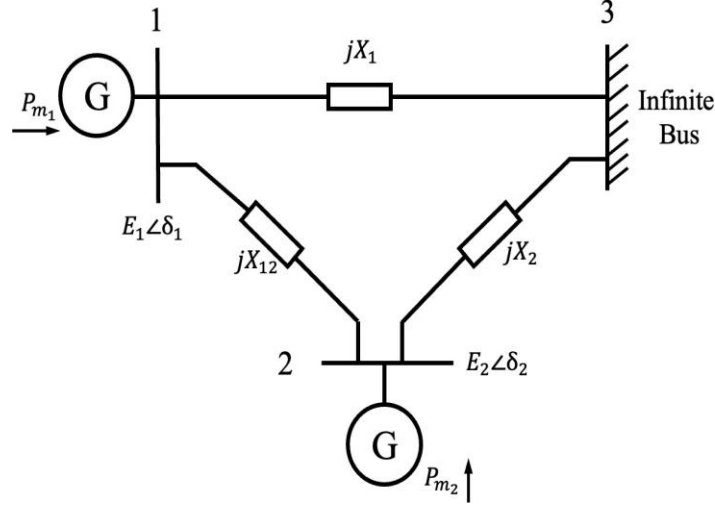


Figure 4.4 TMIB system consists of two generators at buses 1 and 2 connected to the infinite bus at bus 3.

4.2 The Two-Machine Infinite-Bus (TMIB) System

Consider the 4-state system in Figure 4.4 described by

$$\begin{aligned}
 \dot{\delta}_1 &= \omega_1 \\
 M_1 \dot{\omega}_1 &= P_{m1} - D_1 \omega_1 - \frac{E_1}{X_1} \sin(\delta_1) - \frac{E_1 E_2}{X_{12}} \sin(\delta_1 - \delta_2) \\
 \dot{\delta}_2 &= \omega_2 \\
 M_2 \dot{\omega}_2 &= P_{m2} - D_2 \omega_2 - \frac{E_2}{X_2} \sin(\delta_2) - \frac{E_1 E_2}{X_{12}} \sin(\delta_2 - \delta_1)
 \end{aligned} \tag{4.2}$$

with inertia constants $M_1 = M_2 = 6/(2\pi 60) \text{ rad}^{-1} \text{s}^2$, damping constant $D_1 = D_2 = 2/(2\pi 60) \text{ rad}^{-1} \text{s}$, mechanical power inputs $P_{m1} = 1.6$, $P_{m2} = 1.4$ p.u., $X_1 = 0.6$, $X_2 = 0.2$ and $X_{12} = 0.2$ p.u. The generator terminal voltages are equal ($E_1 = E_2 = 1$ p.u.) under normal operating conditions. We consider internal generator faults due to which E_1 and E_2 vary between 0 and 1. We further assume that the pre-fault and the post-fault systems are the same.

To construct $\mathbb{D}$, we densely sampled regions that are more likely to contain fault-on trajectories. Specifically, we sampled points along a few fault-on trajectories with different values of $0 \leq E_1, E_2 < 1$ via `odeint()` in Python. For this example, $\mathbb{D}$ consists of 9000 points from 900 fault-on trajectories with a sampling interval of 0.01 s.

To execute the time-averaged method, we used the same seeding function as we used for the SMIB system. We then evaluated Ψ at points in $\mathbb{D}$ via numerical simulations in Python, and the time-averaged value was computed with a step size of 0.05 s for a time horizon of 20 s. $\bar{\varphi}$ was then obtained by the nearest neighbor algorithm in scikit-learn [42]. For the kernel method, we rearranged $\mathbb{D}$ into sample pairs and scaled them via StandardScaler() in Python. We utilized a Gaussian kernel $\kappa(x_1, x_2) = 0.71\exp(-\|x_1 - x_2\|_2^2/2 \times 0.42^2) + 0.29\exp(-\|x_1 - x_2\|_2^2/2 \times 0.75^2)$. The indicator function was constructed via Algorithm 1 (presented in Chapter 2) with $\gamma = 0.7$.

We next determined the CCT of the faulted scenario where $E_1 = E_2 = 0.4$. As illustrated in Figure 4.5, the CCT of a fault was estimated as the instant when the $\bar{\varphi}$ drops to zero along the fault-on trajectory. We observed that the kernel method and the time-averaged method generate estimates of 0.29 s and 0.28 s, respectively. To find the actual critical clearing time, time-domain simulations were performed in Python for the same fault-on system for varying fault times followed by the post-fault system. The evolution of ω for the fault clearing times of 0.285 s and 0.29 s are plotted in Figure 4.6. The figures indicate that the CCT is closer to 0.29 s, close to the CCTs estimated via the methods we presented.

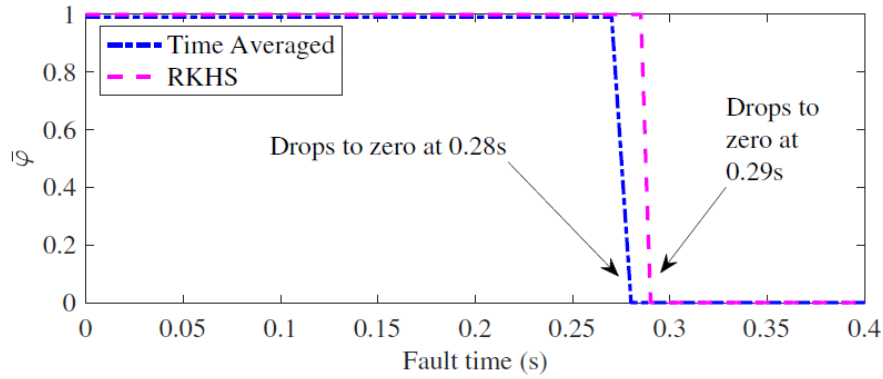


Figure 4.5 The value of the indicator function evaluated along the TMIB fault-on trajectory with $E_1 = E_2 = 0.4$.

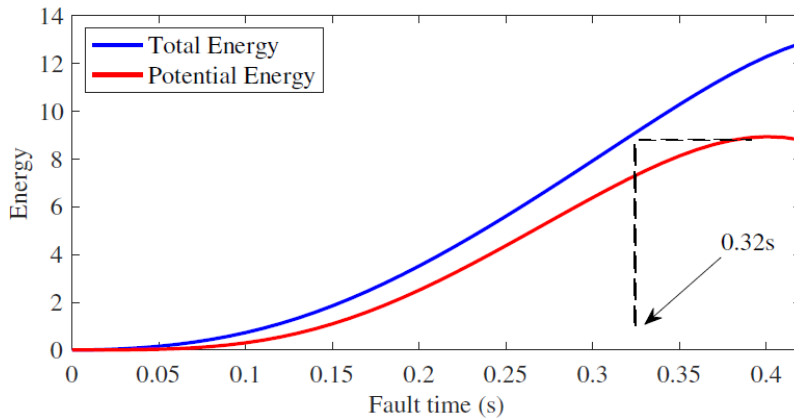


Figure 4.6 The total energy and the potential energy of the TMIB faulted system with $E_1 = E_2 = 0.4$. The CCT is estimated to be 0.32s.

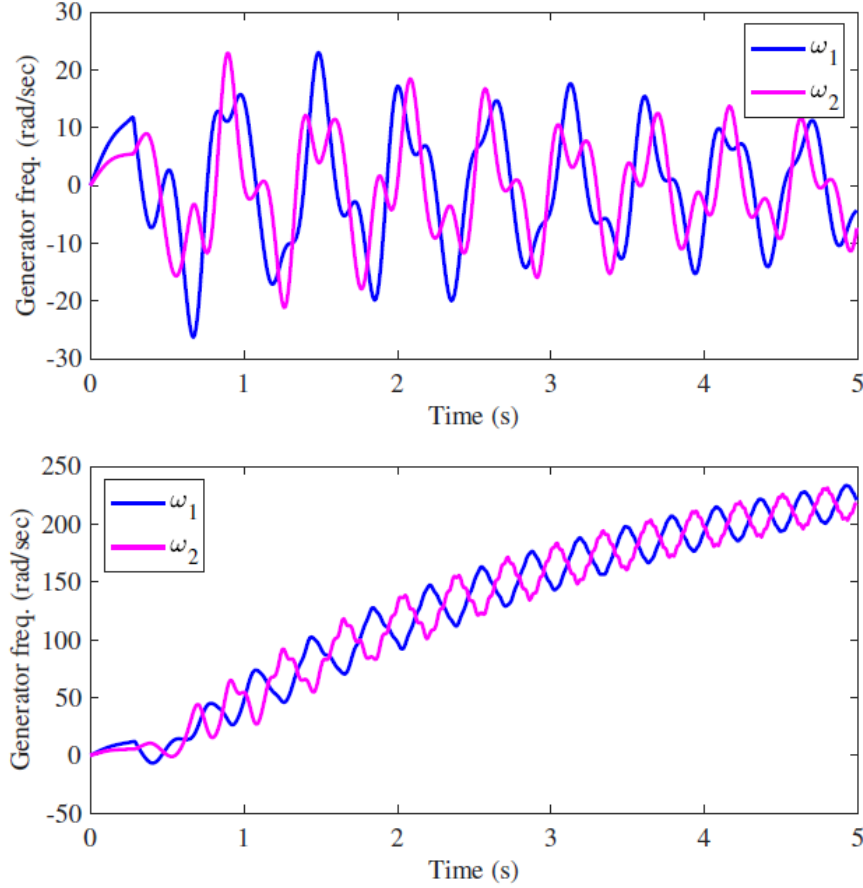


Figure 4.7 Evolution of generator frequencies in the TMIB system with a fault cleared at (a) $t = 0.285$ s and (b) $t = 0.29$ s.

We compared the proposed methods to the standard potential energy boundary surface (PEBS) method in [55]. Different from our data-driven approach, the PEBS method requires an analytic expression of the dynamical system to derive a potential energy function. Figure 4.6 plots the total energy together with the total potential energy of the faulted system with $E_1 = E_2 = 0.4$. The CCT is estimated to be 0.32 s, which is the time when the total energy of the system exceeds the maximum potential energy. These observations suggest that the results from our data-driven approaches better align with the CCT for this system, compared to the PEBS method.

4.3 The Ten-Machine 39-Bus New-England System without IBRs

We now apply the proposed data-driven methods to the ten-machine New England system, as shown in Figure 4.8 [69], conducting simulation analyses both with and without IBR integration. The scenario without IBRs is evaluated in PSSE, where the machines are represented using the classical generator model and the loads are modeled as constant power.

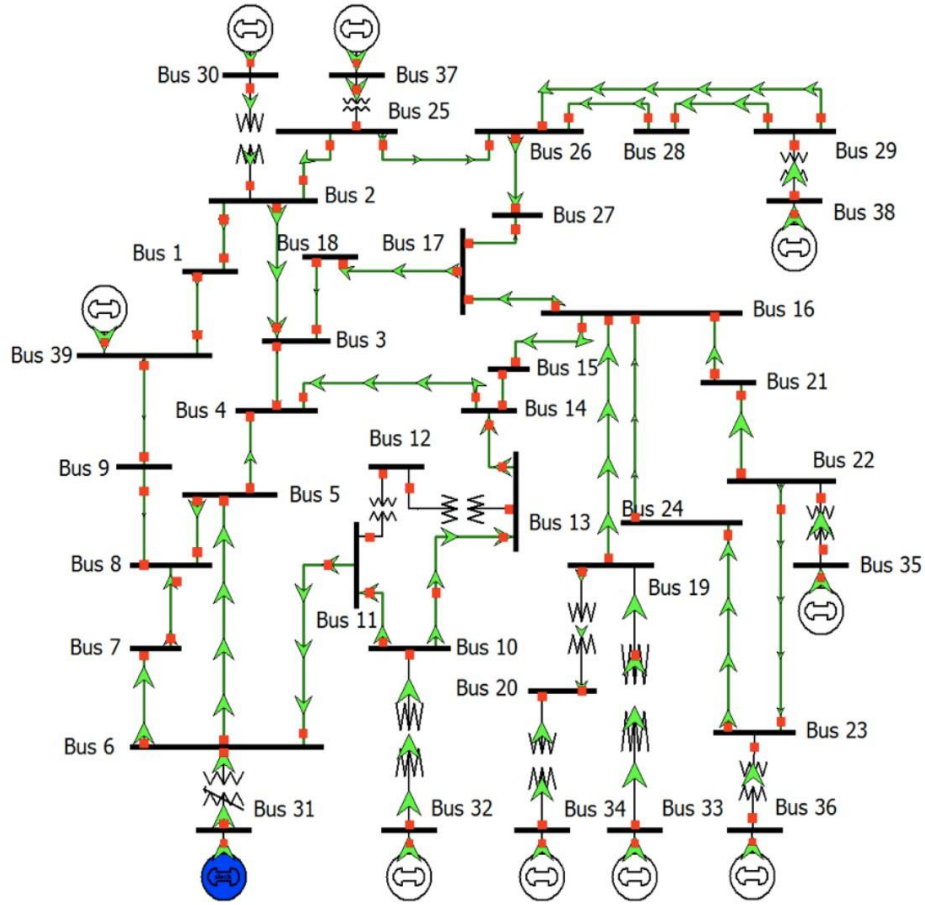


Figure 4.8 The ten-machine New-England Power System (also known as the IEEE 39-bus system).

Table 4.2 Computation times [s] in the different phases for the NE system

Phase	Time-averaged	RKHS
Data Generation	548	23
ROA Indicator Function Construction	9	54
CCT Estimation	0.09	0.12

Table 4.3 Fault information of sample cases studied for the NE system

Case	A	B	C
Fault $Y (\times 10^5)$	$(5-4j)$	$(1-4j)$	$(1-4j)$
Faulted Bus	33	6	24
True CCT [s]	0.29	0.32	0.51
CCT (time-average) [s]	0.29	0.33	0.49
CCT (kernel) [s]	0.28	0.35	0.54

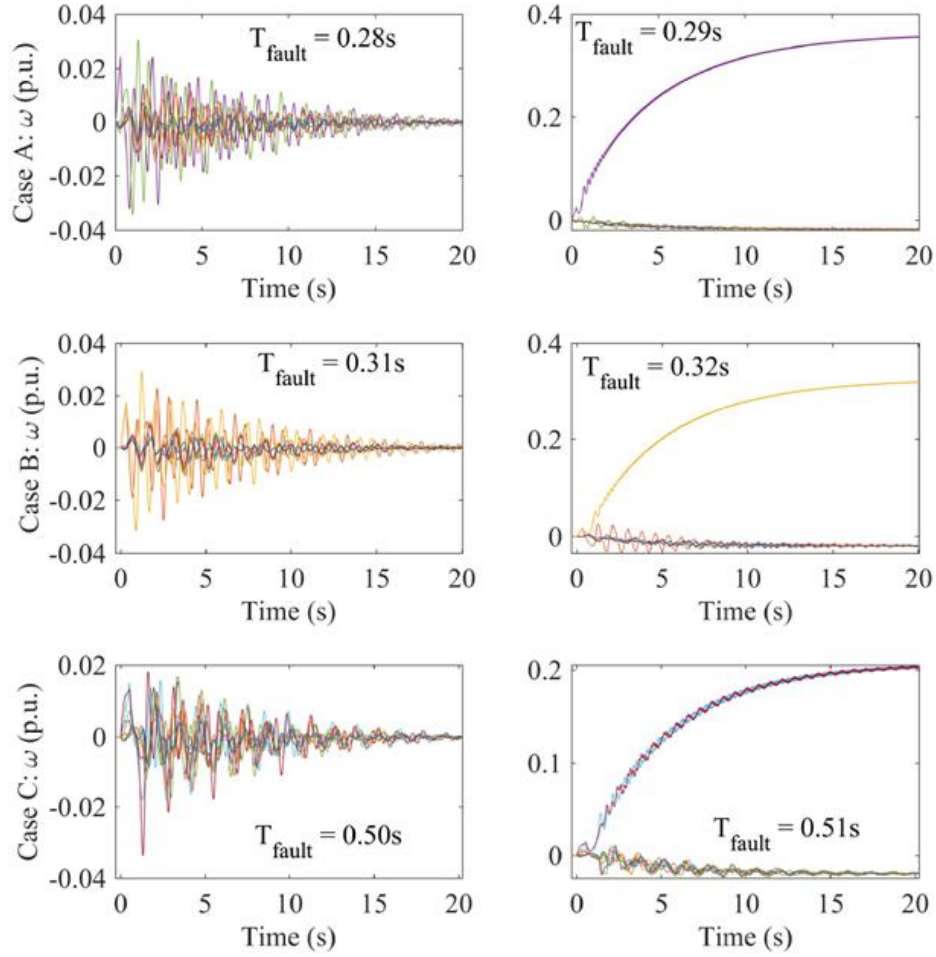


Figure 4.9 The generator ω trajectories for fault scenarios listed in Table 4.3.

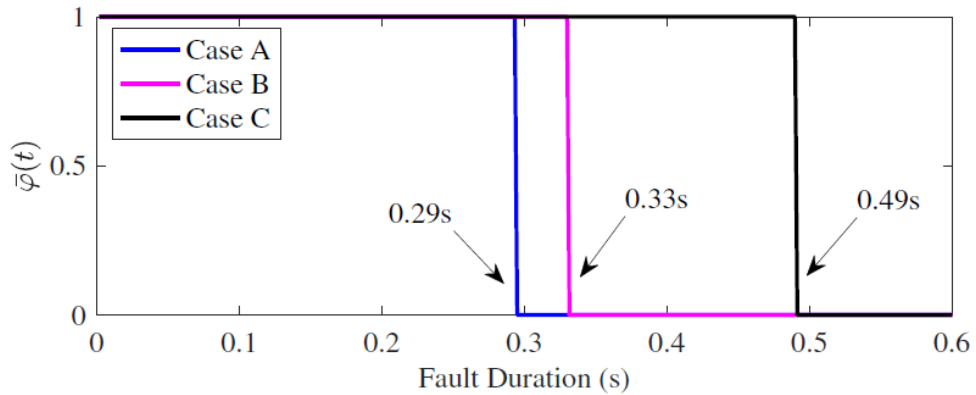


Figure 4.10 The value of the time-averaged indicator function evaluated along fault-on trajectories for faults listed in Table 4.3.

Table 4.3 lists the faults we consider. Assume that the pre- and post-fault systems are the same. We simulated the system in PSSE, and the actual value of CCT was obtained through repeated time-domain simulation implemented in PSSE. To implement the time-averaging approach, we

generated $\mathbb{D}$ by sampling and perturbing points at all 39 buses along five fault-on trajectories with fault admittances being $[10^2 - 10^2j, 10^3 - 10^3j, 10^4 - 10^4j, 10^5 - 10^5j, 10^6 - 10^6j]$ p.u. A total of 150 points along each trajectory are collected and perturbed before adding them to $\mathbb{D}$. Thus, the overall number of points in $\mathbb{D}$ is $39 \times 5 \times 150 = 29,250$. PSSE was then used to simulate the post-fault system from each point in $\mathbb{D}$ with a time horizon of 20 s and a step size of 0.01 s. We used the same seeding function utilized for prior examples. Then, we estimated $\bar{\varphi}$ using the nearest neighbor function from the scikit-learn package in Python.

To implement the kernel method, we sub-sampled 5000 trajectories of length 0.1 s out of the whole dataset used in the time-averaged approach. We then scaled the samples via `StandardScaler()` in Python and used the Gaussian kernel $\kappa(x_1, x_2) = 0.7\exp(-\|x_1 - x_2\|_2^2/2 \times 0.45^2) + 0.2\exp(-\|x_1 - x_2\|_2^2/2 \times 0.7^2) + 0.1\exp(-\|x_1 - x_2\|_2^2/2 \times 2^2)$. We calculated $\bar{\varphi}$ via Algorithm 1 (Chapter 2) with $\gamma = 0.8$ and regularization parameter $\epsilon = 1e^{-13} \times |\mathbb{M}_\gamma|^{-0.2}$.

Figure 4.10 plots the values of the indicator function evaluated along the faulton trajectories listed in Table 4.3. The estimated CCTs are the instants when the indicator function evaluations drop to zero along the trajectories. We compared the estimated CCTs with that obtained from time-domain simulation in Table 4.3. For illustration, we plotted the generator rotor frequency ω for different fault clearing times in Figure 13, which demonstrate the converging/diverging behavior of ω when the fault duration is lesser/greater than the CCTs obtained from the time-domain simulations. Our estimated CCTs are close (within 0.03 s) to that obtained from time-domain simulations for all faults. The methods described are executed on a server equipped with a 3-GHz Intel Xeon 32-core processor. The computation process involves three distinct phases: (a) simulating trajectories using PSSE, (b) analyzing trajectory data to derive an indicator function for the ROA, and (c) using this indicator function to estimate the CCT. The ROA indicator function remains applicable across various faults and perturbations as long as the post-fault system dynamics remain consistent. Therefore, while reducing the time spent on the initial two phases is desirable, the ability to quickly evaluate the ROA indicator function at any point within the state space is crucial.

Table 4.2 summarizes the computational expenses associated with both methods on the 39-bus NE system. The time-averaged method necessitates a substantial number of trajectories compared to the RKHS method to accurately capture the ROA, resulting in an extended data-generation period. Conversely, the RKHS method involves more complexity in the data-analytics phase, as the dictionary needs to be built from the data and the Koopman operator needs to be constructed before the unit eigenfunction, corresponding to the ROA indicator function, can be established. Ultimately, the indicator functions generated by both methods can be evaluated with minimal overhead (≈ 0.1 s) during fault simulations to identify the CCT, thus demonstrating their use for power system planners and operators.

4.4 The Ten-Machine 39-Bus New-England System with IBRs

The Koopman operator theory is applied to the IEEE 39-bus system (illustrated in Figure 4.8), with modifications introduced to assess the impact of IBR integration. The system includes 19 loads, comprising 18 voltage-dependent loads and one composite load model. To represent a weak grid scenario, line impedances are increased, and generator ratings are proportionally reduced with higher levels of IBR penetration. This reduction decreases both power output and system inertia,

thereby heightening the grid’s vulnerability to disturbances.

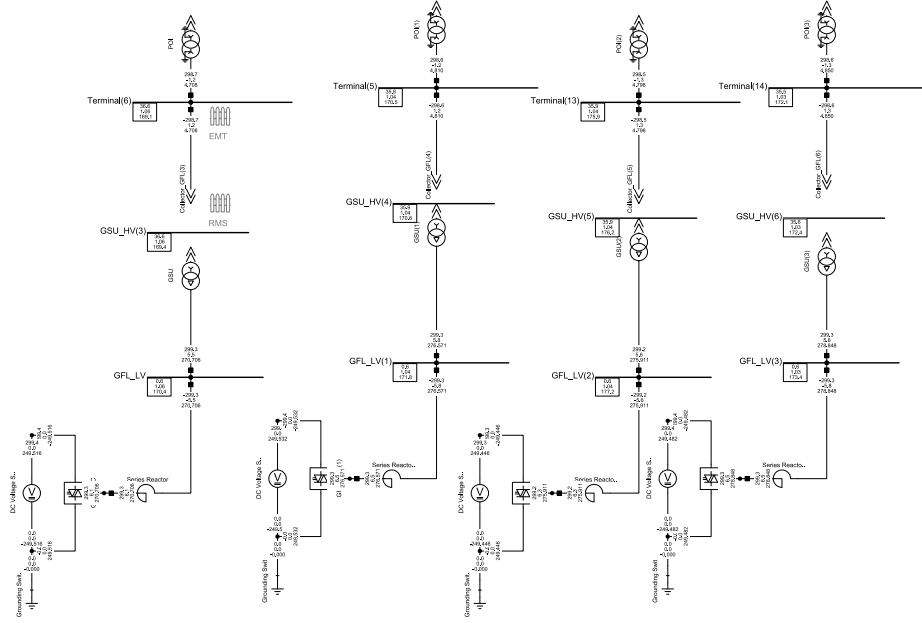


Figure 4.11 IBRs layout in PowerFactory.

The analysis is conducted using the co-simulation framework in PowerFactory, where the IBRs are modeled in the EMT domain and the remainder of the grid is represented in the RMS (three-phase unbalanced) representation. The IBRs are GFL inverters with ride-through settings, providing support during disturbances as described in Chapter 3. The synchronous machines are modeled using Model 2.2 and equipped with an IEEE1A exciter, IEEEGOV1 governor, and PSS1A stabilizer. Despite the availability of multiple machine states, the rotor angle (δ) and speed (ω) are sufficient to determine the system’s CCT. For this analysis, voltage dependent load models are considered. Active powers are represented as constant current and reactive powers as constant voltage load models.

The IBR layout used for the analysis is illustrated in Figure 4.11, where four 300 MW plants are integrated at Buses 17, 24, 27, and 28 of the test system. The co-simulation boundaries are defined at the LV side of the POI transformers. The original IEEE 39-bus system has a total generation capacity of approximately 6200 MW, with Bus 39 serving as both the slack and interconnection bus. To assess the impact of IBR integration, the rating of Bus 39 is reduced by 40%, with IBRs primarily placed in the northeast region. Additionally, line impedances are increased to emulate a weak grid, and generator ratings are reduced to limit fault current contribution. These adjustments significantly modify the system’s power flow characteristics, resulting in IBRs contributing nearly 25% of the total generation.

The data for analysis is generated using a Python-based automated script that interfaces with PowerFactory, with the resulting datasets stored in ‘.csv’ format. Initially, $\mathbb{D}$ is constructed by simulating bus faults at all 39 buses. Only faults located outside the IBR plant layout are considered. Five fault impedances, defined as $Z_{fault} = [10^{-4} + j10^{-12}, 10^{-6} + j10^{-4}, 10^{-8} +$

$j10^{-6}, 10^{-12} + j10^{-12}, 10^{-12} + j10^{-4}] \Omega$ are considered for analysis. The rationale behind this selection is to enable the model to predict the CCT for any impedance within this range. Ten clearing times, spanning from 5 to 20 cycles on a logarithmic scale, are evaluated. Thus, instead of considering 15 uniformly spaced clearing times, we select ten clearing times between 5 (5c) and 20 (20c) cycles (c) on a logarithmic scale, which provides a more representative distribution across the time horizon. This results in a dataset of $39 \times 5 \times 10 = 1950$ simulations.

In the initial simulation of the phase, we noticed that due to fixed clearing times resulted in monotone data for each bus fault. For example, if the clearing times are 5c and 6c, simulations were performed exclusively for 5c across all 39 buses and 5 impedance levels totaling 195 cases with no coverage between 5c and 6c. To ensure broader data coverage, randomness is introduced in the clearing time values within the same 195 simulations, allowing for a more continuous and representative spread. The resulting clearing time (t_{cr}) is given by

$$t_{cr} = t_{ci} + (t_{cj} - t_{ci}) r \quad (4.3)$$

where r is a random number between 0 and 1, t_{ci} and t_{cj} denote the consecutive clearing times. This randomness does not ensure that the point of instability for a specific fault is captured, particularly due to the logarithmic scale employed. However, for the same bus fault, a more evenly distributed dataset is available.

A similar set of simulations is performed for unbalanced faults, including both line-to-line (LL) and line-to-ground (LG) cases. The LL faults may occur on phases AB, BC, or CA, while the LG faults may occur on AG, BG, or CG. Since simulating every possible fault type is computationally intensive, a random selection of unbalanced fault types is considered, resulting in the same total of 1950 simulations for each fault type.

For the simulations, EMT time step of $100 \mu s$ and an RMS time step of $5000 \mu s$ is used, with the co-simulation exchange rate set at the RMS time step. It should be noted that the $\mathbb{D}$ does not include the complete on-fault trajectories; rather, it captures the system states immediately at the fault-clearing time.

Using the PowerFactory co-simulation platform, initialization posed challenges. System states and voltages required considerable time to reach steady state, and applying a fault before this led to instability. As the platform lacks a snap-shot feature to reuse an initialized system, the power flow and initialization process must be repeated for each fault scenario. Therefore, for consistency in the simulations, each fault event is applied at $t = 10$ s, followed by a post-fault simulation of 6 s for analysis.

Using the generated data, the time-averaged unit eigenfunction $\bar{\varphi}$ is estimated using the k-nearest neighbor (KNN) function from the scikit-learn Python package. Different KNN is used for different type of faults. Since the number of unstable scenarios is minimal, initial CCT predictions tend to be optimistic. To address this, a weighted KNN approach is employed, prioritizing unstable cases to yield more conservative CCT estimates.

4.4.1 Balanced Fault Analysis

To begin with, three-phase fault scenarios are analyzed. Since the IBRs are equipped with protection logic, the pre- and post-fault conditions are not assumed to be identical. The simulation data is utilized directly, without explicitly accounting for the IBR operational status, to determine the CCT for each selected fault event. Within dataset $\mathbb{D}$, approximately 37% of the scenarios correspond to unstable cases (i.e., class 0). Consequently, the model is expected to yield a non-conservative estimation.

The initial assessment is carried out using the KNN model with $k=1, 3$ and 5 , all implemented with uniform weighting. The scenarios included in the analysis, along with the estimated (Est.) CCTs obtained for different k values, are presented in Table 4.4. The true CCTs are derived from repetitive time-domain simulations conducted in PowerFactory.

In general, increasing the number of neighbors leads to smoother KNN predictions, as the influence of noise and outliers is reduced. Conversely, with a smaller k , predictions become more sensitive to local variations, with individual outliers exerting stronger influence. Due to the underlying class imbalance, the best CCT estimates are observed with $k = 1$, as this configuration tends to favor unstable cases more strongly.

Table 4.4 Fault information of the test cases studied for 39 bus system

Case	Faulted Bus Number	Z_f [Ω]	Est. CCT ($k = 1$) [s]	Est. CCT ($k = 3$) [s]	Est. CCT ($k = 5$) [s]	True CCT [s]
A	Bus 34	$10^{-10}(2 + 90j)$	0.078	0.078	0.079	0.080
B	Bus 32	$10^{-8}(3000 + 4j)$	0.106	0.106	0.107	0.103
C	Bus 33	$10^{-6}(6 + 2j)$	0.091	0.091	0.094	0.091
D	Bus 19	$10^{-8}(3 + 10j)$	0.075	0.070	0.070	0.082
E	Bus 20	$10^{-12}(1000 + 1j)$	0.075	0.075	0.076	0.0893
F	Bus 16	$10^{-9}(11000 + 55j)$	0.133	0.136	0.138	0.136
G	Bus 10	$10^{-10}(3000 + 6j)$	0.125	0.126	0.132	0.126

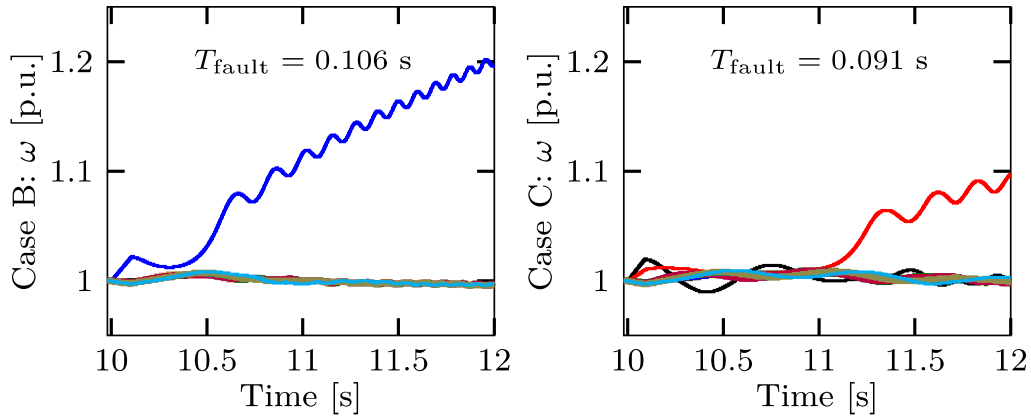


Figure 4.12 Generator trajectories for Case B and C when cleared at estimated CCT.

With $k = 1$, a conservative estimates are noted for Case-A, D, E, F and G. Conservative estimate is observed in Case-A, D, and E with $k = 3$ and 5. Irrespective of the model, an accurate or optimistic result is noted with Cases B and C, thus resulting in instability when cleared with these clearing times (shown in Figure 4.12). Thus, to have conservative estimate for all the scenarios, we adopt weighted KNN (wKNN) approach.

In the developed wKNN framework, the contribution of each neighbor is scaled using user-defined weights rather than the conventional uniform or distance-based schemes. This design choice enables the selective amplification of unstable scenarios while attenuating the influence of less critical cases (i.e., stable scenarios). By appropriately configuring the neighborhood size and the weighting factors, conservative predictions can be achieved, though at the expense of global accuracy. In this study, the neighborhood size ($k = 3$) and weight assignments were empirically tuned such that the prediction error remained below one fundamental cycle (16 ms). a weight assignment of ratio 3:1 is chosen in favor of class 0 (unstable scenarios).

Table 4.5 presents the CCTs obtained using the wKNN approach. Apart from Case B, the method provides conservative estimates for all evaluated scenarios, consistent with the results reported in Table 4.4. Notably, in Case C, the wKNN approach produces a conservative estimate rather than reproducing the true value, thereby illustrating its inherent tendency to prioritize secure operation by avoiding optimistic predictions. An examination of Table 4.5 reveals that the error remains within one fundamental cycle, thereby providing confidence in the reliability of the estimates. Furthermore, a negative error reflects a predicted CCT that exceeds the true value. This may result from missing data samples, imbalanced datasets, or dilution of unstable neighbors in the KNN search. One key observation is that the data generated for Bus 32 in $\mathbb{D}$ do not include clearing time instances in the vicinity of ROA. A potential means of improving the prediction is to incorporate more informative (active) datasets into the KNN training process, thereby enhancing both the accuracy and the conservativeness of the estimates. Even with detailed generator models and IBRs present in the network, the estimated CCTs for most cases remain within a reasonable error of one fundamental cycle, thereby favoring conservative operation and avoiding adverse outcomes.

The corresponding generator ω , obtained by clearing faults at both the estimated and true CCTs, is shown in Figure 4.13. For most of the faults, generator 5 goes out of step, and for faults corresponding to Bus 32 and Bus 10, generator 3 goes out of step. For most fault scenarios, Generator 5 is observed to lose synchronism, while for faults at Bus 32 and Bus 10, Generator 3 goes out of step.

Table 4.5 Estimated CCT using wKNN for three phase faults studied for 39 bus system

Case	Faulted Bus	$Z_f [\Omega]$	Est. CCT [s]	True CCT [s]	Error [ms]
A	Bus 34	$10^{-10}(2 + 90j)$	0.078	0.080	2
B	Bus 32	$10^{-8}(3000 + 4j)$	0.105	0.103	-2
C	Bus 33	$10^{-6}(6 + 2j)$	0.090	0.091	1
D	Bus 19	$10^{-8}(3 + 10j)$	0.068	0.082	14
E	Bus 20	$10^{-12}(1000 + 1j)$	0.074	0.0893	15.3
F	Bus 16	$10^{-9}(11000 + 55j)$	0.132	0.136	4
G	Bus 10	$10^{-10}(3000 + 6j)$	0.124	0.126	2

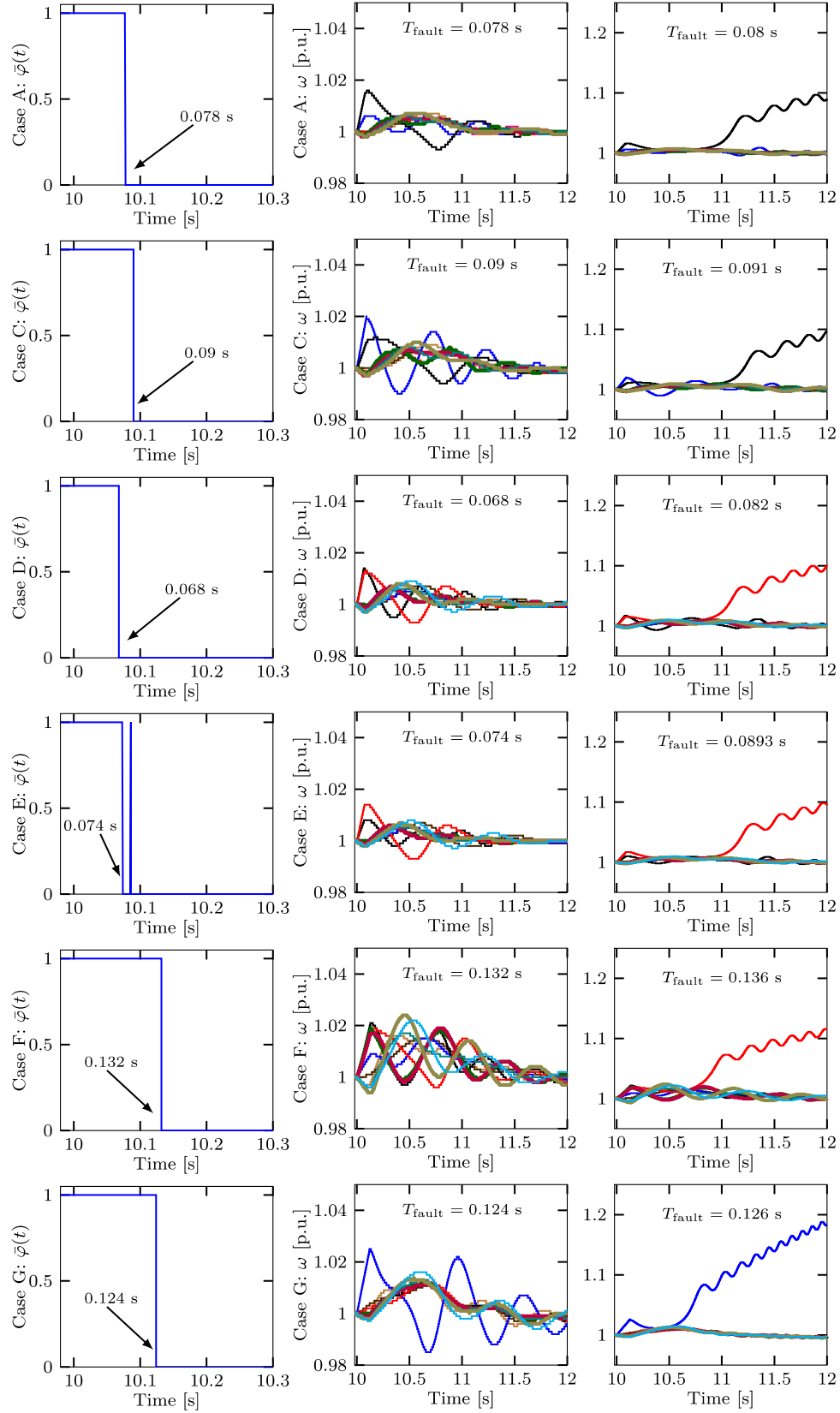


Figure 4.13 Generator trajectories for the faults listed in Table 4.5.

Figure 4.13 also illustrates the evolution of $\bar{\varphi}(t)$ for the fault trajectories listed in Table 4.5. In most cases, the indicator drops sharply to zero at the estimated CCT. In contrast, for Case E, a transient transition is observed in the indicator function. This occurs because the test trajectory passes close to the boundary of the ROA, where the local neighborhood includes both stable and unstable samples. At this point, KNN momentarily assigns a stable label due to the dominance of nearby stable neighbors. As the trajectory progresses further into the unstable region, however, $\bar{\varphi}(t)$ ultimately falls to zero, reflecting the prevalence of unstable neighbors.

To address this issue, post-processing techniques such as simple moving averages or Markov models can be employed to suppress spurious flips near the ROA. Conceptually, this can be viewed as passing the raw predictions through a temporal filter that removes short-lived label reversals. A common approach is to apply a sliding window of size W , within which the majority class label is assigned as the smoothed prediction. For example, $W = 5$ and $\bar{\varphi}(t) = \{0,1,0,0,0\}$ within the window, the smoothed prediction would be 0, reflecting the majority. This yields more stable decision-making and is physically meaningful, as system states do not realistically oscillate between stable and unstable conditions within such a short interval.

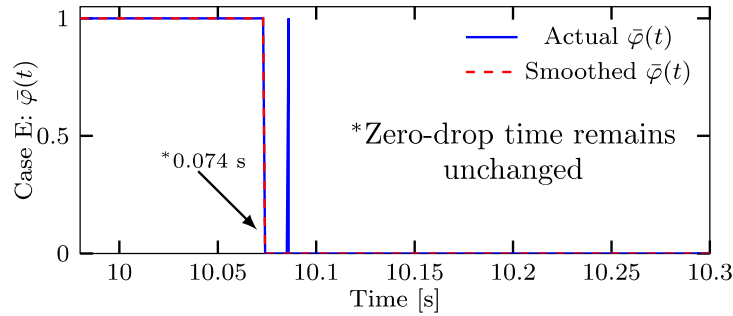


Figure 4.14 Indicator function response by incorporating moving average method.

As illustrated in Figure 4.14, applying this smoothing to Case E eliminates spurious flips without altering the location of the first zero crossing, thereby preserving the conservative nature of the estimate. Prediction smoothing is applied only to faults with clearing times below 150 ms, since most protective devices operate within 5-8 cycles and faults beyond this range are not practically meaningful. For clearing times above 150 ms, only the first zero-crossing instant is considered, irrespective of whether the estimate is conservative or introduces additional error.

Table 4.6 Estimated CCT for two phase faults studied for 39 bus system

Case	Faulted Bus	$Z_f [\Omega]$	Est. CCT [s]	True CCT [s]	Error [ms]
A	Bus 34 (CA)	$10^{-12} (7 + 50000j)$	0.136	0.142	6
B	Bus 32 (CA)	$10^{-6} (3 + 4j)$	0.176	0.177	1
C	Bus 33 (BC)	$10^{-8} (6 + 20000j)$	0.163	0.164	1
D	Bus 19 (BC)	$10^{-9} (300 + 9j)$	0.138	0.142	4
E	Bus 20 (AB)	$10^{-11} (900000 + 1j)$	0.146	0.152	6
F	Bus 16 (AB)	$10^{-10} (10000 + 1j)$	0.230	0.241	11
G	Bus 10 (CA)	$10^{-9} (30 + 4j)$	0.215	0.224	9

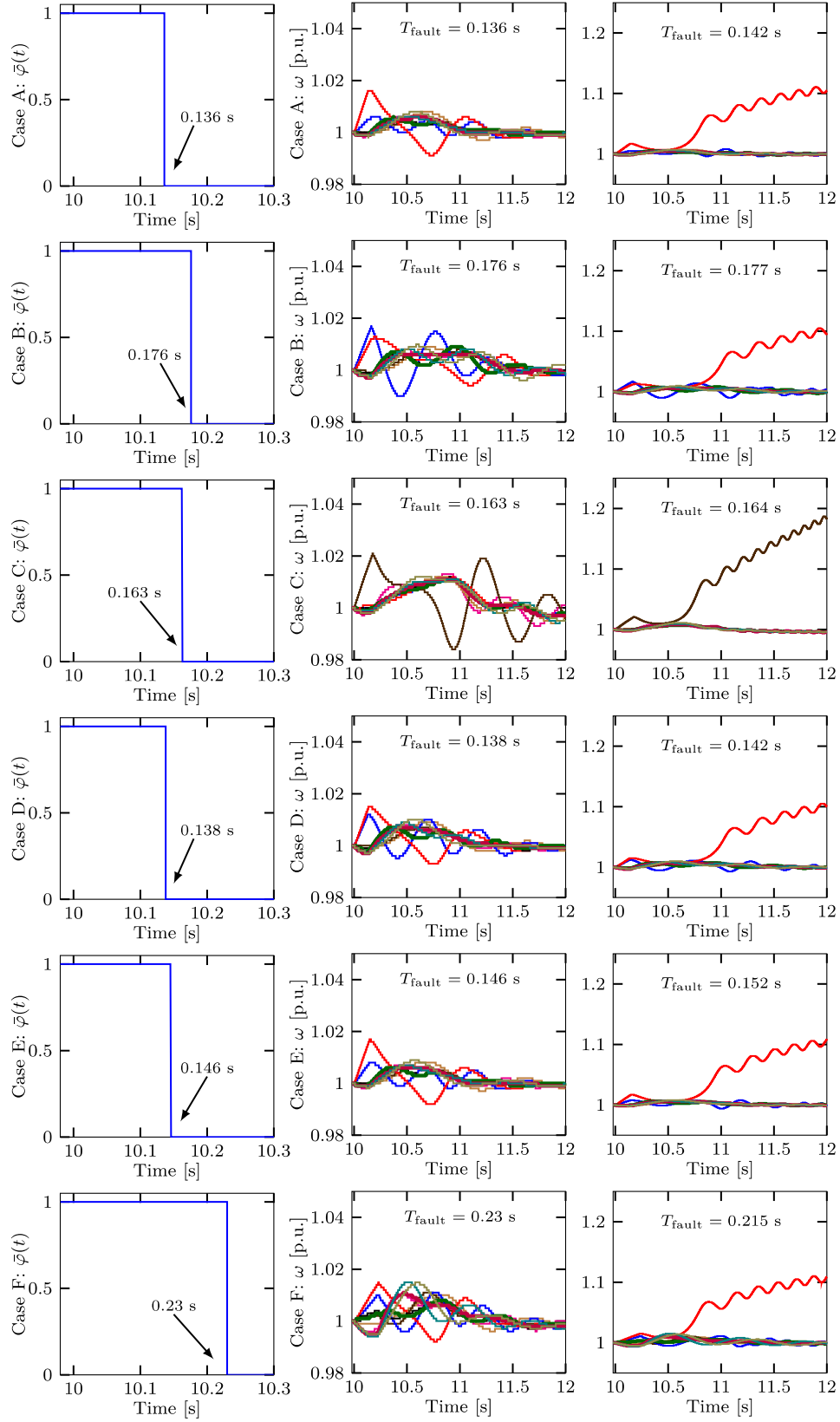


Figure 4.15 Generator trajectories for the faults listed in Table 4.6.

4.4.2 Unbalanced Fault Analysis

A similar analysis is performed for unbalanced faults, including both two-phase and single-phase cases. To construct $\mathbb{D}$, a total of 1819 simulations are carried out, with the unbalanced fault types (AB, BC, or CA) selected randomly. The system state data at the clearing time is recorded and subsequently used to estimate $\bar{\varphi}$ through the wKNN approach. Due to the comparatively lower severity of these fault types, only about 12% of the scenarios exhibit instability.

Table 4.6 summarizes the fault-on trajectories considered in the analysis, with the corresponding fault occurrences indicated in brackets. The wKNN approach consistently produces conservative CCT estimates across all evaluated scenarios, while the true CCT values are obtained through repeated time-domain simulations. As observed in the table, the trained model estimates remain within 16 ms of the true CCTs. Figure 4.15 illustrates the generator trajectories obtained using both the estimated and true CCTs, along with $\bar{\varphi}(t)$. For these faults, no multiple flips of the indicator function are observed, and therefore smoother predictions are not required.

In addition to the LL faults, a similar analysis is conducted for LG faults. In this case, Buses 16, 19, and 20 are observed to become unstable. The KNN model employed for LL faults is used to predict the CCTs of LG faults. The estimated CCTs for LG faults, presented in consistently provide conservative values. The corresponding generator trajectories and estimator function are illustrated in Figure 4.16.

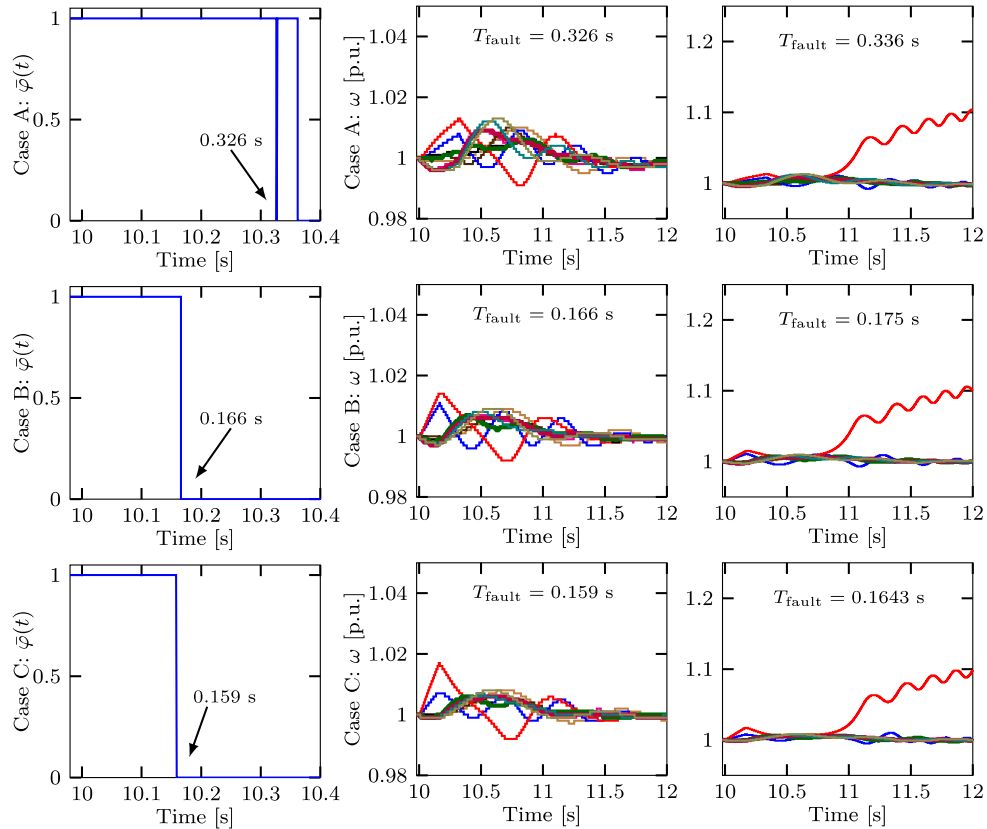


Figure 4.16 Generator trajectories for single phase faults.

Table 4.7 Estimated CCT for single phase faults studied for 39 bus system

Case	Faulted Bus	Z_f [Ω]	Est. CCT [s]	True CCT [s]	Error [ms]
A	Bus 16 (CG)	$10^{-10}(2 + 9j)$	0.326	0.336	10
B	Bus 20 (AG)	$10^{-8}(30000 + 4j)$	0.166	0.175	9
C	Bus 19 (BG)	$10^{-6}(6 + 2j)$	0.159	0.1643	5.3

For Case A, $\bar{\varphi}(t)$ exhibits spurious flips; however, since the CCT exceeds 150 ms, only the first zero-crossing instant is considered, without applying prediction smoothing. Furthermore, although the impedances used for training the models are fixed, evaluations across different impedance ranges demonstrate the robustness and efficacy of the methodology in determining the CCT. This is indeed the case, as the ROA is determined by the underlying system dynamics rather than specific impedance values. By leveraging the Koopman operator, these dynamics are captured in a form that generalizes effectively across different impedance conditions.

Table 4.8 Estimated CCT for all fault types using single wKNN model

Faulted Bus	Type of fault	Est. CCT [s]	True CCT [s]	Error [ms]
Bus 16	3	0.134	0.136	2
	2	0.228	0.241	13
	1	0.326	0.336	10
Bus 19	3	0.070	0.082	12
	2	0.138	0.142	4
	1	0.166	0.175	9
Bus 20	3	0.072	0.0893	17.3
	2	0.146	0.152	6
	1	0.159	0.1643	5.3
Bus 10	3	0.122	0.126	4
	2	0.214	0.224	10
Bus 32	3	0.106	0.103	-3
	2	0.178	0.177	-1
Bus 33	3	0.090	0.091	1
	2	0.162	0.164	2
Bus 34	3	0.076	0.080	4
	2	0.136	0.142	6

Instead of employing separate KNN models for each fault type, a single KNN model is utilized to estimate the CCT. For this purpose, the datasets of three-phase and two-phase faults are combined, resulting in a total of 3,779 points, which are then used to train the wKNN model. The estimated CCTs for the same set of impedances are reported in Table 4.8 where the numbering denotes the fault type. The results show that all estimated CCTs are conservative, except for Bus 32. With the individual models, the estimates for Bus 32 were non-conservative under the three-phase fault and conservative under the two-phase fault, whereas the unified KNN model yields a non-conservative estimate. This suggests that incorporating additional fault scenarios into the dataset may improve the accuracy for this bus. Across all evaluated cases, the estimation errors remain within 16 ms;

however, for Bus 20, the error exceeds 0.6 of a fundamental cycle. The increased error is likely attributable to reduced relevance of the nearest neighbors, as the combined dataset introduces greater variability in the local neighborhoods.

The proposed method demonstrates high reliability, yielding predominantly conservative estimates with only minor deviations into non-conservative predictions. Across different impedance ranges and fault types, the approach achieves conservative estimates in nearly 88% of cases with an average error of about 6 ms. Notably, instead of relying on a wide range of system states, the generator speeds and rotor angles alone are sufficient to capture the transient stability behavior, and the data-driven Koopman operator works precisely even in the IBR rich grid.

4.5 240-bus WECC test system

We extend our data-driven analysis to the 240-bus WECC system, illustrated in Figure 4.17 [70]. The system includes 109 synchronous generators and 37 IBRs. The synchronous machines are modeled using the GENROU dynamic model, supplemented with the SEXS exciter, IEEE ST stabilizer, and a range of governor models (GAST, HYGGOV, and TGOV1). The IBRs are represented using the REGCA1 generator model, REPCB1 electrical control model, and REPCA1 plant control model.

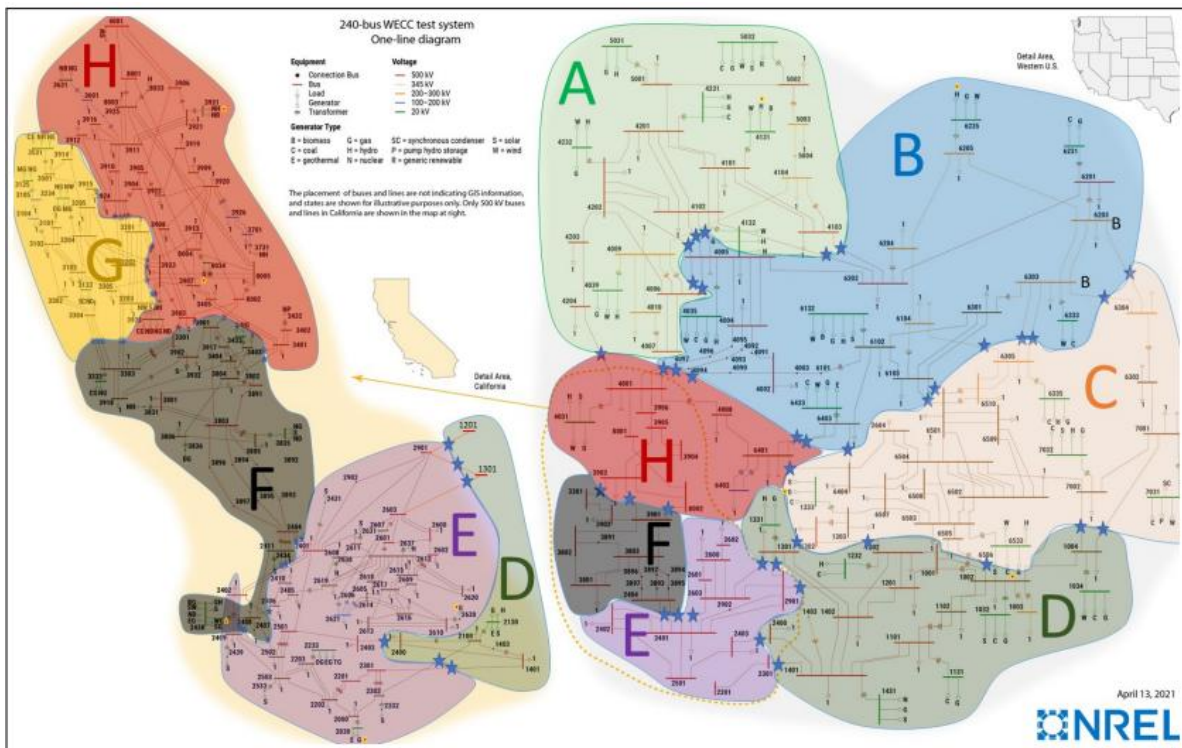


Figure 4.17 Full view of 240-bus WECC system.

Table 4.9 Estimated CCT for 240-bus WECC system

Case	Faulted Bus	$Z_f [\Omega]$	Est. CCT [s]	True CCT [s]	Error [ms]
A	1032	$10^{-11}(20000 + 12j)$	0.136	0.143	7
B	1034	$10^{-7}(80 + 4j)$	0.118	0.127	9
C	1431	$10^{-12}(2 + 900000j)$	0.291	0.3	9
D	1331	$10^{-8}(80000 + 1j)$	0.2496	0.2573	7.7
E	1131	$10^{-10}(400000 + 9j)$	0.289	0.297	8
F	2030	$10^{-4}(5 + 5j)$	0.195	0.206	11
G	2130	$10^{-9}(1 + 3000j)$	0.268	0.271	3

Table 4.9 lists the faults we consider. Assume that the pre- and post-fault systems are the same. We simulated the system in PSSE, and the true CCT was obtained through repeated time-domain simulation implemented in PSSE. To implement the time-averaging approach, we generated $\mathbb{D}$ by creating faults at 55 generator buses (some of which consists of IBR plants also) along five fault-on trajectories with fault impedances being $[10^{-4} + 10^{-12}j, 10^{-6} + 10^{-4}j, 10^{-8} + 10^{-6}j, 10^{-12} + 10^{-12}j, 10^{-12} + 10^{-4}j] \Omega$. A total of 10 clearing instances between 5c and 20c are chosen on a logarithmic scale. Thus, a total of $55 \times 5 \times 10 = 2750$ simulations were carried out. The rotor speed and angles of all the generator buses are recorded at the instant of fault clearing. The wKNN model is trained using these data points.

Table 4.9 summarizes the estimated CCT values for the test scenarios, with indicator functions plotted in Figure 4.18. Consistent with the 39-bus analysis, the evaluation employs a range of impedances not included in training to validate the accuracy of the proposed method. The results indicate that the model produces conservative CCT estimates, with errors of less than one fundamental cycle. Figure 4.19 presents the corresponding generator trajectories (ω) for Cases A through C, comparing faults cleared at both the estimated and true CCTs. As shown, trajectories diverge when faults are cleared at the true CCTs. Although multiple states are available from the synchronous generator and IBR models, rotor angle and speed alone are sufficient for accurate CCT estimation. Incorporating IBR states does not adversely affect the results; however, it increases input dimensionality and can become cumbersome for larger systems.

The developed methodology ensures conservative CCT estimates, regardless of variations in IBR and generator control loops. Using only black-box simulation data and without requiring detailed modeling information, the approach delivers a balanced and dependable solution for transient stability assessment.

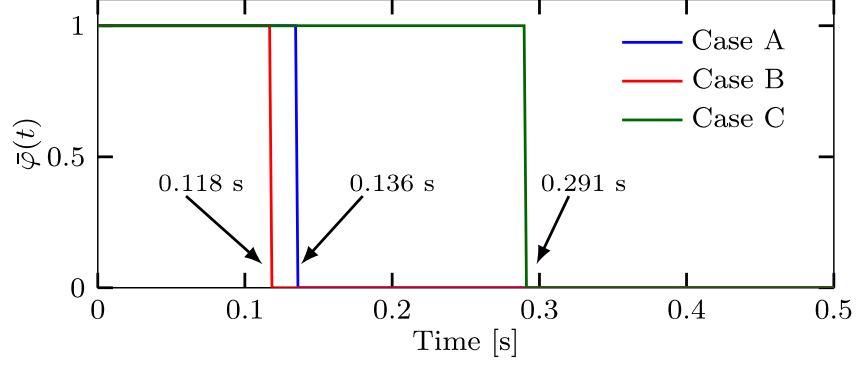


Figure 4.18 The indicator functions obtained for the 240-bus WECC system.

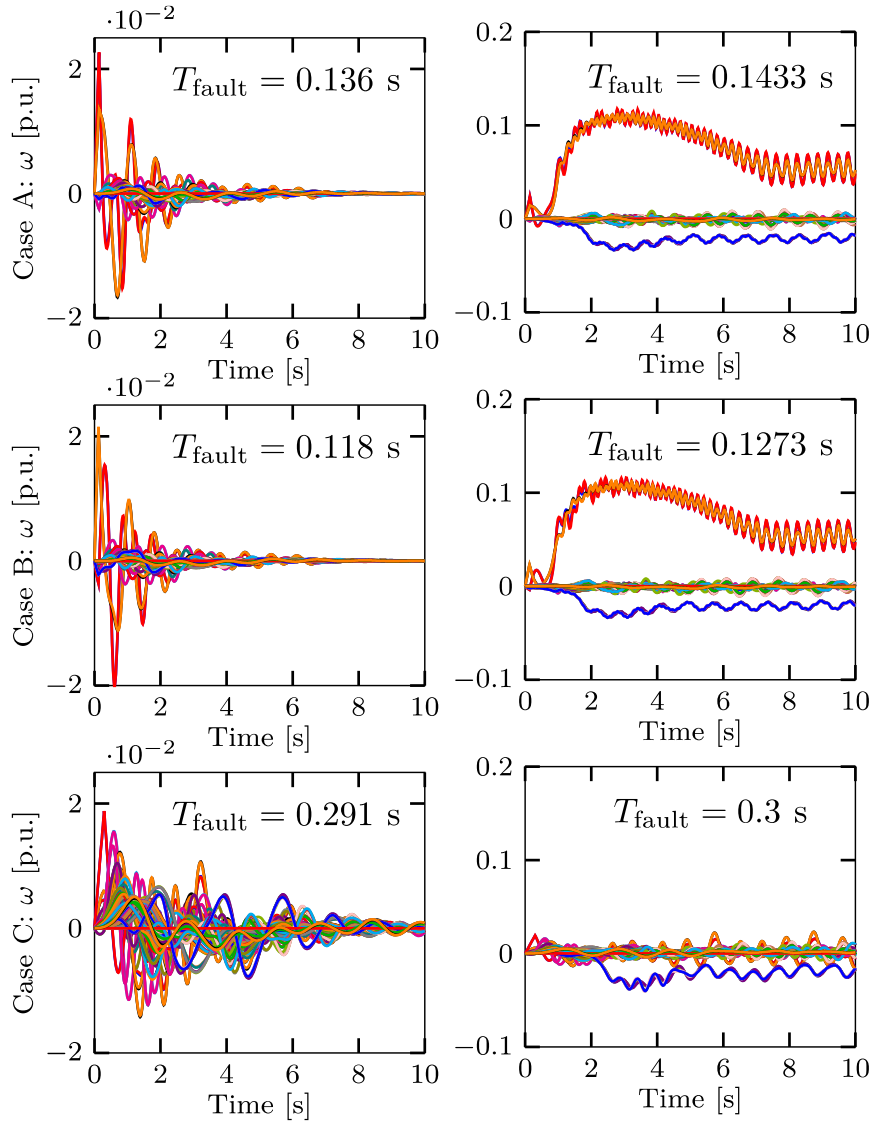


Figure 4.19 Generator trajectories for Case A through C faults listed in Table 4.9.

Table 4.10 Summary of simulation studies conducted across different test systems

Test System	Software Used	Simulations Performed	Comments
IEEE 39-bus (no IBRs)	PSSE	29,250	Balanced three-phase faults
IEEE 39-bus (IBRs)	PowerFactory	1,950 per fault type — total 5,850	Balanced and unbalanced faults considered
240-bus WECC system	PSSE	2,750	Balanced three-phase faults

Table 4.10 provides a brief summary of simulations performed for the test systems considered. The IEEE 39-bus system without IBRs was simulated in PSSE with only balanced three-phase faults considered, resulting in 29,250 cases. For the IEEE 39-bus system with IBRs, simulations were performed in PowerFactory, with 1,950 cases per fault type. Since both balanced and unbalanced faults were included, the total number of simulations reached 5,850. From these simulations, it was observed that faults at generator buses tend to be more severe. Based on this observation, for the 240-bus WECC system we limited the evaluation to generator buses, where IBR plants were also present. For this large-scale system, 2,750 balanced three-phase fault cases were simulated in PSSE. This structured approach ensured methodological consistency while enabling direct comparison between conventional and IBR-rich system behaviors.

4.6 Practical insights and Computation burden

In both conventional and IBR rich power systems, three phase faults remain the most severe type of disturbance as they impose the deepest voltage depression and the greatest overall stress on the network. However, our simulations indicate that in grids with high levels of IBRs the impact of a fault is determined less by its type and more by its location. This is because IBRs provide limited and location dependent fault currents with their response strongly influenced by control and protection settings. As a result, faults occurring at weak grid locations or close to large IBR plants are particularly critical since they expose low short circuit strength and can trigger adverse converter grid interactions. This shows that in modern systems, location sensitivity must be considered as carefully as fault type when assessing transient stability.

From a practical standpoint, the data-driven Koopman framework shows that stability margins such as CCT can be estimated directly from black-box simulators without requiring detailed vendor models, which is critical in IBR-dominated grids. It enables faster contingency screening compared to brute-force simulations, offers flexibility in how data is used, and extends naturally to hybrid EMT–phasor studies that utilities already employ. Validation across different platforms further highlights its potential as a practical and scalable tool for real-world stability assessment.

Compared to brute force time domain simulation, the data driven Koopman framework offers a much more efficient pathway to estimating CCT. Traditional approaches require repeating multiple nonlinear simulations for each new contingency and clearing time which makes wide area studies computationally burdensome and impractical for online use. By contrast, the Koopman method uses an initial set of simulated trajectories to construct a linear operator in a lifted space that captures the system dynamics. Once trained, this operator can be reused across contingencies

allowing new CCT estimates to be obtained directly through linear propagation rather than rerunning full nonlinear simulations. The use of active datasets further improves efficiency since data can be selected or updated strategically to capture the most informative operating conditions, reducing the number of simulations needed while maintaining accuracy.

It is important to note that the Koopman framework does require an upfront investment in generating and training on simulation data. This initial cost can be significant but once the operator is built it can be applied across fault types locations and operating conditions making the marginal cost of additional stability assessments negligible. In practice this reusability is what delivers scalability since the same Koopman model supports many queries at almost no added computational cost.

Both brute force simulation and the Koopman approach can benefit from parallel computing but in different ways. For brute force methods parallelization reduces wall clock time by allowing many contingencies to be simulated at once yet the per contingency burden remains high since each CCT search requires multiple nonlinear runs. For Koopman the generation of training data can also be parallelized and once training is complete inference for new contingencies is essentially instantaneous and does not depend on processor count. This makes the Koopman framework well suited for large scale contingency screening and near real time security assessment in IBR rich grids.

5. Conclusions

In this work, we developed data-driven methods for transient stability analysis in power systems that circumvent the need for explicit analytical equations of system dynamics. We demonstrated that the unit eigenfunction of the Koopman operator can effectively identify the ROA of a stable post-fault equilibrium point. Furthermore, we established a connection between the unit Koopman eigenfunction and numerical integration techniques, which are widely adopted in industry, and leveraged this property to estimate the CCT of a fault.

To estimate the unit eigenfunction from simulation data, we proposed two complementary approaches: the time-averaged method and the kernel method. The time-averaged method requires longer simulation trajectories but is computationally lightweight, whereas the kernel method operates with shorter snapshot bursts but entails a more involved computation.

A key advantage of the proposed framework is its explicit reliance on data rather than detailed system models. This is particularly valuable in IBR dominated systems, where components are often represented as input–output black-box models and little to no information about internal dynamics is available. By utilizing only a black-box simulator, the developed methods enable practitioners to evaluate transient stability in such settings. We demonstrated that the methodology extends naturally to hybrid co-simulation power system models, highlighting its versatility and practical relevance for modern grids. Finally, the efficacy of the proposed approach is validated through four power system case studies simulated across different platforms.

5.1 Future Work

The present work evaluates transient stability assessment of the modern power grids using data-driven methods. Based on this analysis, the following future research directions are proposed:

1. As IBR integration increases, utilities face a growing need to employ full-scale EMT simulations to assess grid stability. A key direction of this work is to evaluate the performance and adaptability of the proposed methods when applied to EMT data. While the current study focused on CCT estimation using RMS and hybrid models, EMT-based validation was not performed. Given the high necessity of EMT simulations, it is important to investigate the adaptability of the data-driven methods and determine the level of additional system detail required. Ultimately, this should help to determine whether full-scale EMT is necessary, or whether appropriately tuned hybrid models are sufficient for reliable stability assessment.
2. The present study focused on CCT estimation under a 25% IBR penetration level. Future work should aim to develop nomograms that capture the influence of varying penetration levels and IBR control parameters on system responses to fault events. Such insights would provide grid operators with practical tools to ensure secure and reliable grid operation.
3. Future work will focus on using linear tools in lifted models to better map stability regions. This will help identify how the ROA evolves with operating conditions and fault events, providing deeper insights into system security. In particular, ROA boundary sensitivity with respect to transfer flows and IBR injections will be analyzed to determine stability limits for critical contingencies under fixed fault clearing times. Additionally, examining

the eigenvectors of key functions will help reveal interactions among system states, offering deeper insight into the dynamics that govern stability.

4. Up to this point, the stability study has focused on GFL inverters. Future work should incorporate GFM controls and investigate how the interaction between GFMs and GFLs affects data-driven stability assessment. In addition, the role of GFM states in enhancing stability characterization should be evaluated, with particular attention to scenarios approaching a 100% IBR-based grid, where IBR-driven instabilities may emerge.
5. Another promising direction is to study how IBR control loop parameters influence stability assessment. New datasets reflecting different control specifications will be generated and combined with existing data to evaluate their effect on CCT estimation. The objective is to construct a parameterized data-driven model that incorporates controller details, enabling more accurate and flexible stability assessments.

References

- [1] U.S. Energy Information Administration, [New solar plants expected to support most U.S. electric generation growth - U.S. Energy Information Administration \(EIA\)](#).
- [2] Midcontinent Independent System Operation (MISO), “Renewable Integration Impact Assessment”: [RIIA Summary Report520051.pdf](#).
- [3] Joint NERC & WECC Staff Report, “San Fernando Disturbance Report”, November 2020. Available at: [Report](#).
- [4] Joint NERC & WECC Staff Report, Odessa Disturbances Report, “Odessa Disturbances Report”, September 2021. Available at: [Report](#).
- [5] Y. Cheng *et al.*, "Real-World Subsynchronous Oscillation Events in Power Grids With High Penetrations of Inverter-Based Resources," in *IEEE Transactions on Power Systems*, vol. 38, no. 1, pp. 316-330, Jan. 2023, doi: 10.1109/TPWRS.2022.3161418.
- [6] EMT Modeling and Simulation – Who Needs an EMT Model for Doing Stability Studies? <https://www.esig.energy/event/webinar-emt-modeling-and-simulation-who-needs-an-emt-model-for-doing-stability-studies/>
- [7] NERC, “DRAFT White Paper: EMT Modeling Adoption for Interconnection and Planning Studies”. Available at: [Report](#).
- [8] Q. -H. Wu, Y. Lin, C. Hong, Y. Su, T. Wen and Y. Liu, "Transient Stability Analysis of Large-scale Power Systems: A Survey," in *CSEE Journal of Power and Energy Systems*, vol. 9, no. 4, pp. 1284-1300, July 2023, doi: 10.17775/CSEEJPES.2022.07110.
- [9] P. Varaiya, F. F. Wu, and R.-L. Chen, “Direct methods for transient stability analysis of power systems: Recent results,” *Proceedings of the IEEE*, vol. 73, no. 12, pp. 1703–1715, 1985.
- [10] M. A. Pai, “Survey of practical direct methods of stability analysis in power systems,” *Electric machines and power systems*, vol. 9, no. 2-3, pp. 131–143, 1984.
- [11] P. W. Sauer, M. A. Pai, and J. H. Chow, *Power system dynamics and stability: with synchrophasor measurement and power system toolbox*. John Wiley & Sons, 2017.
- [12] A.-A. Fouad and V. Vittal, *Power system transient stability analysis using the transient energy function method*. Pearson Education, 1991.
- [13] H.-D. Chiang, *Direct methods for stability analysis of electric power systems: theoretical foundation, BCU methodologies, and applications*. John Wiley & Sons, 2011.
- [14] M. Pai, *Power system stability: analysis by the direct method of Lyapunov*. North-Holland, 1981, vol. 3.
- [15] ———, *Energy function analysis for power system stability*. Springer Science & Business Media, 1989.
- [16] H. K. Khalil, “Nonlinear systems third edition,” *Patience Hall*, vol. 115, 2002.
- [17] P. A. Parrilo, *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. California Institute of Technology, 2000.
- [18] Y. Zou, M.-H. Yin, and H.-D. Chiang, “Theoretical foundation of the controlling uep method for direct transient-stability analysis of network- preserving power system models,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 50, no. 10, pp. 1324– 1336, 2003.
- [19] R. T. Treinen, V. Vittal, and W. Kliemann, “An improved technique to determine the controlling unstable equilibrium point in a power system,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 43, no. 4, pp. 313–323, 1996.

- [20] H.-d. Chiang and J. S. Thorp, "The closest unstable equilibrium point method for power system dynamic security assessment," in *26th IEEE Conference on Decision and Control*, vol. 26. IEEE, 1987, pp. 72–76.
- [21] C.-W. Liu and J. S. Thorp, "A novel method to compute the closest unstable equilibrium point for transient stability region estimate in power systems," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 7, pp. 630–635, 1997.
- [22] H.-D. Chiang, J. Tong, and Y. Tada, "On-line transient stability screening of 14,000-bus models using tepco-bcu: Evaluations and methods," in *IEEE PES General Meeting*. IEEE, 2010, pp. 1–8.
- [23] N. Kakimoto, "Transient stability analysis of electric power system via lur'e type Lyapunov function," *Trans. IEE of Japan, E*, vol. 98, no. 5, pp. 63–79, 1978.
- [24] M. Anghel, F. Milano, and A. Papachristodoulou, "Algorithmic construction of Lyapunov functions for power system stability analysis," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 9, pp. 2533–2546, 2013.
- [25] W. Tan and A. Packard, "Stability region analysis using polynomial and composite polynomial Lyapunov functions and sum-of-squares programming," *IEEE Transactions on Automatic Control*, vol. 53, no. 2, pp. 565–571, 2008.
- [26] G. Franze, D. Famularo, and A. Casavola, "Constrained nonlinear polynomial time-delay systems: A sum-of-squares approach to estimate the domain of attraction," *IEEE transactions on automatic control*, vol. 57, no. 10, pp. 2673–2679, 2012.
- [27] H. Choi, P. J. Seiler, and S. V. Dhople, "Robust power systems stability assessment with sum of squares optimization," in *2015 IEEE Power & Energy Society General Meeting*. IEEE, 2015, pp. 1–5.
- [28] North American Electric Reliability Corporation, "Electromagnetic Transient Modeling for BPS Connected Inverter-Based Resources—Recommended Model Requirements and Verification Practices," in *NERC Reliability Guidelines*, March 2023.
- [29] J. James, D. J. Hill, A. Y. Lam, J. Gu, and V. O. Li, "Intelligent time-adaptive transient stability assessment system," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 1049–1058, 2017.
- [30] F. Li, Q. Wang, Y. Tang, and Y. Xu, "An integrated method for critical clearing time prediction based on a model-driven and ensemble cost-sensitive data-driven scheme," *International Journal of Electrical Power & Energy Systems*, vol. 125, p. 106513, 2021.
- [31] M. Senyuk, M. Safaraliev, F. Kamalov, and H. Sulieman, "Power system transient stability assessment based on machine learning algorithms and grid topology," *Mathematics*, vol. 11, no. 3, p. 525, 2023.
- [32] X. Wu, X. Wang, X. Wang, P. E. Caines and J. Liu, "Comparative Analysis of Learning Based Methods for Transient Stability Assessment," 2024 56th North American Power Symposium (NAPS), El Paso, TX, USA, 2024, pp. 1-6, doi: 10.1109/NAPS61145.2024.10741857.
- [33] H. Choi, R. Elliott, D. Trudnowski, and D. Venkat, "Transient stability quantification of power systems with inverter-based resources via koopman operator based machine learning approach," *Electric Power Systems Research*, vol. 237, p. 111035, 2024.
- [34] N. Mohammed, W. Zhou, D. Ramasubramanian, B. Bahrani, S. Dutta, and M. Bello, "Data-driven estimation of impedance of inverter-based resources for efficient stability evaluation," *Electric Power Systems Research*, vol. 244, p. 111559, 2025.

- [35] H. Wang, P. Sun, J. Sahebkar Farkhani, and Z. Chen, “A data-driven identification method for impedance stability analysis of inverter-based resources,” *IET Smart Grid*, vol. 7, no. 5, pp. 572–582, 2024.
- [36] B. O. Koopman and J. v. Neumann, “Dynamical systems of continuous spectra,” *Proceedings of the National Academy of Sciences*, vol. 18, no. 3, pp. 255–263, 1932.
- [37] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, “A data-driven approximation of the koopman operator: Extending dynamic mode decomposition,” *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.
- [38] J. H. Tu, “Dynamic mode decomposition: Theory and applications,” Ph.D. dissertation, Princeton University, 2013.
- [39] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.
- [40] B. Hou, S. Bose, and U. Vaidya, “Sparse learning of kernel transfer operators,” in *2021 55th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2021, pp. 130–134.
- [41] B. Hou, S. Sanjari, N. Dahlin, and S. Bose, “Compressed decentralized learning of conditional mean embedding operators in reproducing kernel hilbert spaces,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, 2023, pp. 7902–7909.
- [42] B. Hou, A. R. R. Matavalam, S. Bose, and U. Vaidya, “Propagating uncertainty through system dynamics in reproducing kernel hilbert space,” *Physica D: Nonlinear Phenomena*, p. 134168, 2024.
- [43] A. Lasota and M. C. Mackey, *Chaos, fractals, and noise: stochastic aspects of dynamics*. Springer Science & Business Media, 1998, vol. 97.
- [44] U. Vaidya and P. G. Mehta, “Lyapunov measure for almost everywhere stability,” *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 307–323, 2008.
- [45] Mezi’c, “Spectral properties of dynamical systems, model reduction and decompositions,” *Nonlinear Dynamics*, vol. 41, no. 1, pp. 309–325, 2005.
- [46] A. Mauroy and I. Mezic, “Global stability analysis using the eigenfunctions of the koopman operator,” *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3356–3369, 2016.
- [47] M. Budisic, R. Mohr, and I. Mezic, “Applied koopmanism,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 4, p. 047510, 2012.
- [48] Y. Susuki and I. Mezi’c, “Nonlinear koopman modes and power system stability assessment without models,” *IEEE Transactions on Power Systems*, vol. 29, no. 2, pp. 899–907, 2013.
- [49] Y. Susuki, I. Mezic, F. Raak, and T. Hikihara, “Applied koopman operator theory for power systems technology,” *Nonlinear Theory and Its Applications, IEICE*, vol. 7, no. 4, pp. 430–459, 2016.
- [50] Y. Xu, M. Netto, and L. Mili, “Propagating parameter uncertainty in power system nonlinear dynamic simulations using a koopman operator-based surrogate model,” *IEEE Transactions on Power Systems*, vol. 37, no. 4, pp. 3157–3160, 2022.
- [51] B. Hou, S. Sanjari, N. Dahlin, S. Bose, and U. Vaidya, “Sparse learning of dynamical systems in rkhs: An operator-theoretic approach,” in *Proceedings of the 40th International Conference on Machine Learning, ICML 2023*, 2023.
- [52] L. Di Natale, Y. Lian, E. T. Maddalena, J. Shi, and C. N. Jones, “Lessons learned from data-driven building control experiments: Contrasting gaussian process-based mpc, bilevel deepc, and deep reinforcement learning,” in *2022 IEEE 61st conference on decision and control (CDC)*. IEEE, 2022, pp. 1111–1117.

- [53] A. Mauroy and I. Mezić, “On the use of fourier averages to compute the global isochrons of (quasi) periodic dynamics,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 3, 2012.
- [54] S. Klus, I. Schuster, and K. Muandet, “Eigendecompositions of transfer operators in reproducing kernel hilbert spaces,” *Journal of Nonlinear Science*, vol. 30, no. 1, pp. 283–315, 2020.
- [55] P. W. Sauer and M. A. Pai, *Power system dynamics and stability*, 2006.
- [56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [57] E. Yeung, S. Kundu, and N. Hodas, “Learning deep neural network representations for koopman operators of nonlinear dynamical systems,” in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 4832–4839.
- [58] S. P. Nandanoori, S. Sinha, and E. Yeung, “Data-driven operator theoretic methods for global phase space learning,” in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 4551–4557.
- [59] Z. Li, D. Meunier, M. Mollenhauer, and A. Gretton, “Optimal rates for regularized conditional mean embedding learning,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 4433–4445, 2022.
- [60] B. Hou, S. Sanjari, A. Koppel, and S. Bose, “Compressed online learning of conditional mean embedding,” *arXiv preprint arXiv:2405.07432*, 2024.
- [61] L. Song, J. Huang, A. Smola, and K. Fukumizu, “Hilbert space embeddings of conditional distributions with applications to dynamical systems,” in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 961–968.
- [62] Klebanov, I. Schuster, and T. J. Sullivan, “A rigorous theory of conditional mean embeddings,” *SIAM Journal on Mathematics of Data Science*, vol. 2, no. 3, pp. 583–606, 2020.
- [63] C. Richard, J. C. M. Bermudez, and P. Honeine, “Online prediction of time series data with kernels,” *IEEE Transactions on Signal Processing*, vol. 57, no. 3, pp. 1058–1067, 2008.
- [64] T. Athay, R. Podmore, and S. Virmani, “A practical method for the direct analysis of transient stability,” *IEEE Transactions on Power Apparatus and Systems*, no. 2, pp. 573–584, 1979.
- [65] PowerWorld Corporation, *PowerWorld Simulator*. [Online]. Available at: [Models Associated with Generators](#).
- [66] X. Gao, D. Zhou, A. Anvari-Moghaddam and F. Blaabjerg, “Grid-Following and Grid-Forming Control in Power Electronic Based Power Systems: A Comparative Study,” *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*, Toronto, ON, Canada, 2021, pp. 1-6.
- [67] P. Rodriguez, J. Pou, J. Bergas, J. I. Candela, R. P. Burgos and D. Boroyevich, "Decoupled Double Synchronous Reference Frame PLL for Power Converters Control," in *IEEE Transactions on Power Electronics*, vol. 22, no. 2, pp. 584-592, March 2007, doi: 10.1109/TPEL.2006.890000.
- [68] “IEEE Standard for Interconnection and Interoperability of Inverter-Based Resources (IBRs) Interconnecting with Associated Transmission Electric Power Systems,” in *IEEE Std 2800-2022*, vol., no., pp.1-180, 22 April 2022, doi: 10.1109/IEEESTD.2022.9762253.

- [69] Illinois Center for a Smarter Electric Grid (ICSEG), “IEEE 39-bus system,” 2023. [Online]. Available: <https://icseg.iti.illinois.edu/ieee-39-bus-system/>.
- [70] Wang, Bin, Richard Wallace Kenyon, and Jin Tan. 2022. Developing a PSCAD Model of the Reduced 240-Bus WECC Test System: Preprint. Golden, CO: National Renewable Energy Laboratory NREL/CP-6A40-82287.<https://www.nrel.gov/docs/fy22osti/82287.pdf>.