Analysis of Model Free Predictors for Interface Signal Delay Compensation in Real-Time Cosimulation

Elutunji Buraimoh, *Member, IEEE*, Gokhan Ozkan, *Senior Member, IEEE*, Laxman Timilsina, *Member, IEEE*, Grace Muriithi, *Graduate Student Member, IEEE*, Behnaz Papari, *Senior Member, IEEE*, Ali Arsalan, *Graduate Student Member, IEEE*, Ali Moghassemi, *Graduate Student Member, IEEE*, Mustafa Ozden, *Member, IEEE*, and Christopher Edrington, *Senior Member, IEEE*

***Abstract*—Real-time cosimulation of geographically dispersed laboratories enables extensive system simulations but faces significant challenges from communication delays, impacting accuracy and stability. This issue is crucial in real-time power system cosimulation, where delays can disrupt synchronism and hinder dynamic analyses. This article proposes model-free predictive delay compensation methods as viable alternative signal transformation-based methods. This study explores the frequency domain stability of a model-free framework for delay prediction and compensation at power system interfaces using the ideal transformer method as interface algorithm. Similarly, time-domain implementation reveals signal amplitude magnification, addressed by introducing a delay- and frequency-dependent normalizing factor. This framework adapts interface coupling signals, enabling real-time parameter tuning without complex processing or system models, enhancing co-simulation accuracy and stability for distributed power system analyses.**

***Index Terms*—Delay, delay compensation, delay prediction, ideal transformer method, interface algorithm (IA), power system co-simulation.**

I. Introduction

POWER systems real-time distributed cosimulation requires network protocols to enable communication between geographically dispersed digital simulators. Strict synchronization of these simulators ensures consistent outcomes across all participating laboratories. Distributed cosimulation offers advantages, such as efficient and cost-effective testing of complex systems, eliminating the need for centralized facilities, supporting collaboration and resource sharing, and creating realistic testing environments that enhance real-world performance and safety [1]. However, real-time distributed cosimulation faces significant challenges, particularly with communication latency [2]. Latency between signal transmission and the recipient's numerical integration can cause instability and inaccuracies in networked environments [3].

Power systems are especially vulnerable to such delays, which can cause synchronization issues and reduce simulation fidelity, making them unsuitable for dynamic and transient studies. Addressing communication latency is crucial in real-time cosimulation, particularly in complex systems, such as power grids, where even minimal delays can significantly impact accuracy and reliability. Real-time distributed cosimulation depends on high-bandwidth, low-latency communication networks to ensure real-time information exchange between simulators. Effective synchronization algorithms are essential for maintaining consistent results across locations [4]. As technology advances, virtual integration for real-time digital simulators is expected to expand across industries. Solutions to mitigate latency include enhanced communication networks, clock synchronization, parallel processing, predictive algorithms, and specialized cosimulation frameworks [5], [6].

Real-time distributed cosimulation divides the system under investigation into subsystems, simulated by separate real-time simulators that exchange interface coupling signals of the interface algorithm (IA). The precision of a power system cosimulation IA is essential for high fidelity [7]. The IA mitigates communication network effects by conserving energy at the coupling point, defining the strategy for connecting subsystems. Selecting an IA is critical for cosimulation stability [8]. Various strategies have been developed for power hardware-in-the-loop (PHIL) systems, suitable for distributed cosimulation due to similar system partitioning and real-time needs. In PHIL, an adaptive power coupling element interacts between virtual and physical domains, using an IA to reduce instabilities and errors from coupling components, such as power amplifiers and probes [9]. This enables stable, realistic power system stability studies [10].

Unlike PHIL, which relies on instantaneous voltage and current waveforms, real-time distributed cosimulation IAs

Received 25 November 2024; accepted 5 January 2025. Date of publication 28 January 2025; date of current version 4 April 2025. This work was supported by Clemson University's VIPR-GS, under Cooperative Agreement with the US Army DEVCOM GVSC, under Grant W56HZV-21-2-0001. Paper no. TII-24-6303. *(Corresponding author: Elutunji Buraimoh.)*

The authors are with the Holcombe Department of Electrical and Computer Engineering and Department of Automotive Engineering, Clemson University, Clemson, SC 29634 USA (e-mail: eburaim@clemson.edu).

Digital Object Identifier 10.1109/TII.2025.3528548

transform these waveforms into slowly varying quantities exchanged between simulators. This approach addresses the challenges posed by time-varying delays, reordering, and packet loss in assigned network communication. The sending end transforms current and voltage waveforms, and the receiving end reconstructs these signals to maintain system stability and accuracy [1]. Traditional transformation methods for delay compensation in real-time distributed cosimulation include root mean square (rms), dynamic phasors (DPs), and synchronous reference frames (SRF). RMS methods decompose voltages and currents into their rms values, frequency, and phase angle for transmission and reconstruction [11]. Although these method compensates for time delays, they struggle with maintaining fidelity during transients and require high computational complexities [12]. DPs use time-varying Fourier coefficients to represent system quantities, including nonstationary behaviors in power system simulations and compensating for delays through phase shifts [13]. The SRF method, which targets the accurate validation of control algorithms and embedded software, overcomes some limitations of phasor interfaces [14].

Each method requires careful consideration of the specific application and subsystem characteristics involved. The choice of IA significantly impacts the simulation's accuracy, speed, and stability, underscoring the importance of selecting the appropriate technique based on the cosimulation scenario's requirements [1]. Existing real-time distributed cosimulation IAs rely on complex signal processing techniques to mitigate communication delays, introducing computational burdens and limiting applicability to nonpower system variables. To address these limitations, a model-free framework for predicting and compensating delays in cosimulated systems offers a promising solution that can be applied to IAs like the ideal transformer model (ITM). This framework leverages an adaptive, parameter-tuning predictor system to predict and compensate for delays without complex signal transformations or human interference. This advancement enhances cosimulation accuracy and stability, enabling reliable dynamic and transient analyses in geographically dispersed power system models.

This article contributes to the field of real-time power system cosimulation, specifically in the area of interconnecting geographically separated real-time simulators with accurate time synchronization and alignment as pointed out in literature [15], delay compensation, signal integrity, and preservation of simulation fidelity in geographically dispersed laboratories. Some contribtution are given as follows.

1) This work implements a model-free framework for predicting and compensating communication delays in real-time power system cosimulation. This framework does not require complex signal transformations, making it more efficient and easier to implement while eliminating complex computation.
2) In this article, we propose a look-up table-based normalization technique to address the challenges of amplitude distortion due to communication delays and coupling signal frequency. This method enhances the accuracy of predicted signals, ensuring better alignment with the original signal characteristics.
3) The proposed prediction systems' effectiveness is validated using coupling signals from an ITM partitioned IEEE 5-bus system with IEEE 33-bus system. These signals were transmitted between two geographically separated laboratories without signal transformation, demonstrating the practical applicability of the framework especially for dynamic studies.
4) This work evaluates the performance of the prediction system in comparison to traditional methods, such as DP and SRF. It assesses real-time coupling errors and examines the effectiveness of the prediction system in minimizing these errors, as well as residual power injection or generation and frequency discrepancies within subsystems.

The rest of this article is organized as follows. Section II provides the details of the proposed model-free framework for delay compensation, along with the principles of ITM used for the experimental validation. Section III provides the frequency domain analysis of the prediction systems and the impact of the tuning parameters and delays on the stability of the prediction system. Section IV presents the results of the validation experiments of the proposed prediction in comparison with the traditional methods. Finally, Section V concludes this article.

II. ITM Coupling Signal and Prediction Design

IAs ensure accurate interactions between subsystems in real-time cosimulation. The ITM is a widely used IA in power system cosimulation, facilitating the exchange of coupling signals—voltage and current—between subsystems [16]. Fig. 1 shows the ITM interface's configuration in a connected power system cosimulation. Predictors in each subsystem process delayed signals $v(t-\tau)$ and $i(t-\tau)$, generating predicted signals $\hat{v}(t)$ and $\hat{i}(t)$ to mitigate delays. A controlled voltage source imposes the measured voltage from one subsystem onto another, while a controlled current source mirrors current behavior [17]. These sources ensure accurate electrical representation across subsystems, maintaining fidelity.

Coupling signals—voltage and current—are essential for dynamic interactions between subsystems. Ideally, the voltage signal $v(t)$ from Subsystem 1 is imposed on Subsystem 2, maintaining consistent voltage behavior. Similarly, the current signal $i(t)$ from Subsystem 2 is imposed on Subsystem 1, ensuring consistent current behavior. These signals, transmitted through a network, introduce delays affecting accuracy and stability. Delay compensation mechanisms are therefore crucial. The open-loop transfer function for ITM is shown in (1), where $Z_1(s)$ and $Z_2(s)$ represent the network impedances of subsystems 1 and 2, respectively

$$G_0 = \frac{Z_1(s)}{Z_2(s)} e^{-s\Delta t}. \tag{1}$$

The delay signal prediction compensates for communication delay by predicting the signal's state when it reaches the remote simulator. Predictive algorithms estimate the future state based on current and past signals, compensating for transmission time [18]. These predictions are integrated into controlled sources, ensuring accurate signal imposition, and mitigating delays. ITM maintains high fidelity without altering electrical conditions, excelling in steady-state error performance and

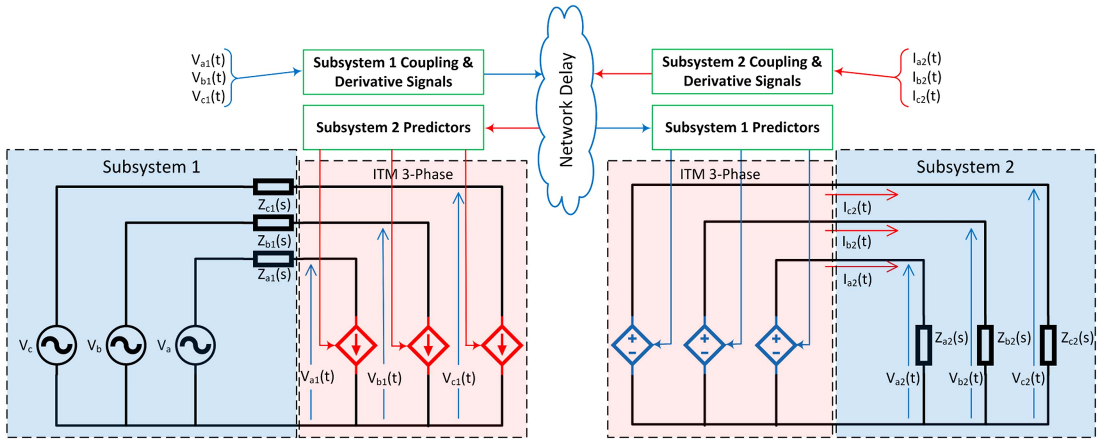

Fig. 1. Configuration of the ITM interface with integrated predictors for delay compensation in a closed-loop power system cosimulation.

stability [1]. However, ITM's sensitivity to delays can lead to instability during transient events. Implementing delay prediction enhances cosimulation accuracy and stability, addressing these challenges and improving reliability.

Since communication delay (τ) can be estimated by synchronizing clocks on both sides of the network and subtracting packet send timestamps from receipt timestamps, the known delay assumption holds true and real-time value can be estimated in the two subsystems. This is because τ can be measured. For mathematical definition, prediction errors $e_v(t)$ and $e_i(t)$, representing the difference between the undelayed remote signal and its local prediction, are shown in (2) and (3). Equations (4) and (5) define the coupling error $c_v(t)$ and $c_i(t)$, the error between the undelayed remote signal and its delayed local instance during the time period. The goal of the one-parameter predictor is to asymptotically reduce prediction error to zero and keep it below the coupling error

$$e_v(t) = v(t) - \hat{v}(t) \tag{2}$$

$$e_i(t) = i(t) - \hat{i}(t) \tag{3}$$

$$c_v(t) = v(t) - v(t-\tau) \tag{4}$$

$$c_i(t) = i(t) - i(t-\tau). \tag{5}$$

Consider a single-input system with a nonlinear structure, represented by (6) where $x(t)$ and $u(t)$ denote the system state vector and control input, respectively. This model provides insight into dynamic characteristics, essential for analyzing control strategies

$$\dot{x}(t) = f(x(t)) + g(x(t)) \cdot u(t) \tag{6}$$

Equation (6) describes the state-space dynamics of the nonlinear system, where $\dot{x}(t)$ is the rate of change of the system state over time. The nonlinear functions $f(x(t))$ and $g(x(t))$ define internal dynamics and input interactions. $f(x(t))$ captures disturbances or uncertainties, bounded by a known continuous function of $x(t)$. The control gain $g(x(t))$ has a known sign and is constrained by identifiable state functions.

To simplify control, sliding mode control transforms a high-order nonlinear system into a reduced first-order system by introducing a sliding surface s, defined by follows:

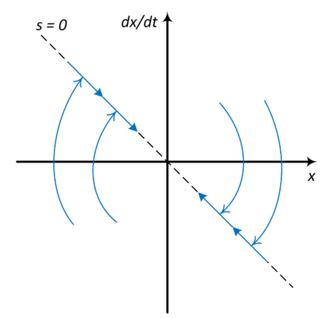

Fig. 2. Sliding mode control principle.

$$s = \left(\frac{d}{dt} + \alpha\right) x = 0. \tag{7}$$

The objective is to ensure $x(t)$ reaches and stays on this surface, achieving desired performance as shown in Fig. 2. The control input $u(t)$ is designed for two phases: Reaching phase: Drive the state to $s = 0$; and Sliding phase: Keep the state on the surface, stabilizing dynamics. The control law is derived by substituting the sliding surface condition into the dynamics, yielding $u(t)$ in the following equation:

$$u(t) = -\frac{f(x(t)) + \alpha x}{g(x(t))}. \tag{8}$$

This control input ensures the system slides along the surface from (7), transforming the nonlinear dynamics of (6) into first-order behavior. Applying sliding mode control drives errors $e_v(t)$ and $e_i(t)$ to zero or to a minimized surface. The sliding mode control law with parameter α is defined in (9) and (10). These surfaces correspond to control deviations for voltage and current signals between simulators, minimizing errors for synchronized behavior. Sliding mode control's robustness to parameter variations and disturbances makes it suitable for fields, such as robotics, aerospace, and power systems [19]. Ensuring the system slides on a predetermined surface guarantees stability and performance despite uncertainties

$$\frac{d[e_v(t)]}{dt} = -\alpha e_v(t) \tag{9}$$

$$\frac{d[e_i(t)]}{dt} = -\alpha e_i(t). \tag{10}$$

This describes the coupling signal error dynamics under sliding mode control. The predictor relies on first-order time-delay system dynamics, shown in (11) and (12), leading to the prediction error dynamics in (15). It outlines the time derivatives of estimated voltage $\hat{v}(t)$ and current $\hat{i}(t)$ based on delayed measurements $v(t-\tau)$ and $i(t-\tau)$. The terms $\alpha[v(t-\tau) - \hat{v}(t-\tau)]$ and $\alpha[i(t-\tau) - \hat{i}(t-\tau)]$ act as corrections for differences between delayed and predicted signals

$$\frac{d[\hat{v}(t)]}{dt} = \frac{d[v(t-\tau)]}{dt} + \alpha[v(t-\tau) - \hat{v}(t-\tau)] \tag{11}$$

$$\frac{d[\hat{i}(t)]}{dt} = \frac{d[i(t-\tau)]}{dt} + \alpha[i(t-\tau) - \hat{i}(t-\tau)]. \tag{12}$$

To analyze these equations in the frequency domain, we apply the Laplace transform. The time derivative $\frac{d}{dt}$ becomes multiplication by s in the Laplace domain, and the time delay τ introduces a factor of $e^{-\tau s}$, as follows:

$$\begin{aligned}\left[V(s) - \hat{V}(s)\right] \cdot s + \left[V(s) - \hat{V}(s)\right] \cdot \alpha e^{-\tau s} \\ = s \cdot \left[V(s) - e^{-s\tau} V(s)\right]\end{aligned} \tag{13}$$

$$\begin{aligned}\left[I(s) - \hat{I}(s)\right] \cdot s + \left[I(s) - \hat{I}(s)\right] \cdot \alpha e^{-\tau s} \\ = s \cdot \left[I(s) - e^{-s\tau} I(s)\right].\end{aligned} \tag{14}$$

Thus, (15) is derived from (13) and (14) to analyze these dynamics

$$\frac{E_v(s)}{C_v(s)} = \frac{E_i(s)}{C_i(s)} = \frac{s}{s + \alpha e^{-\tau s}}. \tag{15}$$

The single-parameter predictor's performance is limited by its single design degree of freedom, α. Thus, a second parameter, β, is introduced to enhance the design, as follows:

$$\frac{d[e_v(t)]}{dt} = -\alpha e_v(t) - \beta \frac{d[e_v(t)]}{dt} \tag{16}$$

$$\frac{d[e_i(t)]}{dt} = -\alpha e_i(t) - \beta \frac{d[e_i(t)]}{dt}. \tag{17}$$

This extension converts the system dynamics from a time-delay to a neutral form without changing the predictor's order, as shown in (18) and (19). With β set to zero, the two-parameter predictor reverts to the one-parameter design. No additional inputs or measurements are required, ensuring compatibility with the original system and maintaining performance

$$\begin{aligned}\frac{d[\hat{v}(t)]}{dt} = \frac{d[v(t-\tau)]}{dt} + \alpha[v(t-\tau) - \hat{v}(t-\tau)] \\ + \beta \frac{d[v(t-\tau) - \hat{v}(t-\tau)]}{dt}\end{aligned} \tag{18}$$

$$\begin{aligned}\frac{d[\hat{i}(t)]}{dt} = \frac{d[i(t-\tau)]}{dt} + \alpha[i(t-\tau) - \hat{i}(t-\tau)] \\ + \beta \frac{d[i(t-\tau) - \hat{i}(t-\tau)]}{dt}.\end{aligned} \tag{19}$$

The Laplace transform is applied to both sides to analyze the equations in the frequency domain

$$\begin{aligned}\left[V(s) - \hat{V}(s)\right] \cdot s + \left[V(s) - \hat{V}(s)\right] \cdot \beta s e^{-\tau s} \\ + \left[V(s) - \hat{V}(s)\right] \cdot \alpha e^{-\tau s} \\ = s \cdot \left[V(s) - e^{-\tau s} V(s)\right]\end{aligned} \tag{20}$$

$$\begin{aligned}\left[I(s) - \hat{I}(s)\right] \cdot s + \left[I(s) - \hat{I}(s)\right] \cdot \beta s e^{-\tau s} \\ + \left[I(s) - \hat{I}(s)\right] \cdot \alpha e^{-\tau s} \\ = s \cdot \left[I(s) - e^{-\tau s} I(s)\right].\end{aligned} \tag{21}$$

This term modifies the transfer function for a refined error estimation by accounting for the rate of change

$$\frac{E_v(s)}{C_v(s)} = \frac{E_i(s)}{C_i(s)} = \frac{s}{s + \beta s e^{-\tau s} + \alpha e^{-\tau s}}. \tag{22}$$

This transfer function represents the dynamics of the two-parameter predictor, enhancing performance and stability by considering both delayed signal and delayed rate of change signal. It is used to evaluate predictor performance and stability.

To find the expression for the predicted signals $\hat{v}(t)$ and $\hat{i}(t)$, we need to integrate both sides of (11) and (12) to achieve (23) and (24) for one parameter

$$\hat{v}(t) = v(t-\tau) + \alpha \int [v(t-\tau) - \hat{v}(t-\tau)]\, dt \tag{23}$$

$$\hat{i}(t) = i(t-\tau) + \alpha \int \left[i(t-\tau) - \hat{i}(t-\tau)\right] dt. \tag{24}$$

In addition, we integrate both sides of (18) and (19) to achieve (25) and (26) for two parameters, with respect to time

$$\hat{v}(t) = v(t-\tau) + \frac{\alpha}{1+\beta} \int [v(t-\tau) - \hat{v}(t)]\, dt \tag{25}$$

$$\hat{i}(t) = i(t-\tau) + \frac{\alpha}{1+\beta} \int \left[i(t-\tau) - \hat{i}(t)\right] dt. \tag{26}$$

This shows how the predicted signal depends on the delayed input, the control parameters α and β, and the accumulated error over time.

III. Stability Analysis

To study stability in one parameter prediction, we substitute $s = j\omega$ (where $j = \sqrt{-1}$ and ω is the angular frequency) into the root of (15): $j\omega + \alpha e^{-j\omega\tau} = 0$. Equating real and imaginary parts leads to conditions on ω and α for stability. Solving these conditions, we find that for the system to be asymptotically stable, α must satisfy (27). Thus, stability requires that α remains within this bound based on the delay τ

$$0 < \alpha < \frac{\pi}{2\tau}. \tag{27}$$

Equation (15) is expressed as a standard transfer function with complex frequency to obtain the frequency response. To find

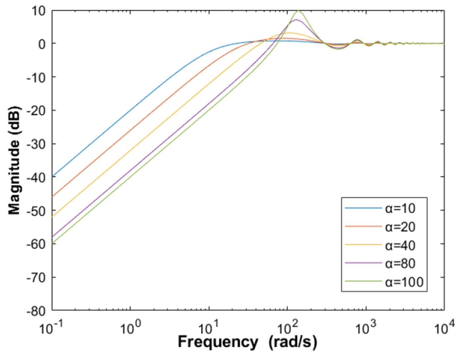

Fig. 3. Frequency response of the one-parameter predictor with a 10 ms delay for various α values.

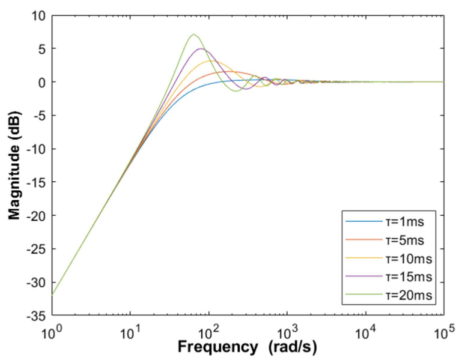

Fig. 4. Frequency response of the one-parameter predictor with $\alpha = 40$ for different delays.

the magnitude of the frequency response $M(j\omega)$, we rewrite the denominator using $\alpha e^{-j\omega\tau} = \alpha(\cos(\omega\tau) - j\sin(\omega\tau))$ and compute the magnitude as the ratio of the numerator and denominator magnitudes. This results as follows:

$$M(j\omega) = \frac{|\omega|}{\sqrt{\omega^2 - 2\omega\alpha\sin(\omega\tau) + \alpha^2}}. \tag{28}$$

Here, $M(j\omega)$ represents the response magnitude, capturing the effect of prediction or state tracking error (numerator) and communication-induced coupling error (denominator). Ideally, the state tracking error is zero, mitigating delay effects. Selecting the design parameter α is crucial for ensuring predictor stability and minimizing $e_v(t)$ and $e_i(t)$. In the frequency domain, the transfer function gain in (15) must stay below one across all frequencies to attenuate coupling errors $c_v(t)$ and $c_i(t)$. If the gain exceeds one at any frequency, the coupling error may be amplified, compromising predictor performance despite stability.

Figs. 3 and 4 show the Bode plots for the frequency response of (11) under different delay estimates τ and design parameter α. All plotted pairs meet the stability criterion. Figs. 3 and 4 illustrate essential performance properties of the one-parameter predictor. Fig. 3 shows that higher α values improve steady-state tracking at low frequencies but may cause peaks over 0 dB at certain ranges. At higher frequencies, the predictor's error reduction effectiveness decreases, as the magnitude approaches 0 dB, indicating a transfer function near one. Fig. 4 highlights that increasing communication delay τ reduces the frequency range where the magnitude stays below 0 dB, diminishing the predictor's effectiveness.

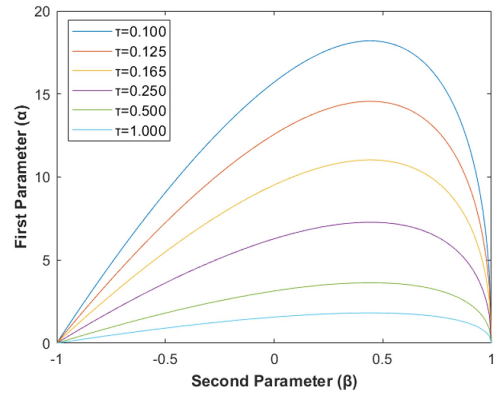

Fig. 5. Stability boundaries of the tuning parameters α and β as a function of delay τ for a two-parameter predictor system.

The stability limits for design parameters α and β are derived based on the estimated delay τ. The characteristic dynamics from (11), (12), (18), and (19) are analyzed. The characteristic equation from (22) is: $s + \beta s e^{-\tau s} + \alpha e^{-\tau s} = 0$. To simplify, we multiply by $e^{\tau s}$ and substitute $s = j\omega$, where ω is the frequency, and use Euler's formula to expand the delay terms. By separating real and imaginary parts, we derive stability conditions for α and β, linking them to ω within the interval $[0, \pi/\tau]$. Further simplification with trigonometric identities yields the open-loop stability boundary as follows:

$$\alpha = \frac{1}{\tau}\arccos(-\beta)\sqrt{1-\beta^2}. \tag{29}$$

Equation (29) relates the stability parameters α, β, and delay τ. Fig. 5 shows the open-loop stability boundaries for different delay values τ in a two-parameter predictor. As delay increases, these boundaries shrink, limiting the design space for maintaining stability. This indicates that the double-parameter design faces challenges similar to the single-parameter design, with larger delays. However, adding the parameter β increases the maximum allowable α for stability, as shown in Fig. 5, where α reaches its threshold at the same β value for any $\tau > 0$.

From (22), the frequency response is derived by substituting $s = j\omega$, where j is the imaginary unit and ω is the angular frequency. Using Euler's identity, $e^{-j\omega\tau} = \cos(\omega\tau) - j\sin(\omega\tau)$, we separate the real and imaginary parts in the denominator. The real part is $\alpha\cos(\omega\tau) - \beta\omega\sin(\omega\tau)$, and the imaginary part is $\omega(1 + \beta\cos(\omega\tau)) - \alpha\sin(\omega\tau)$. The magnitude $M(j\omega)$ is given by the ratio of the numerator to the denominator's magnitude, resulting as follows:

$$M(j\omega) = \frac{|\omega|}{\sqrt{(1 + \beta^2 + 2\beta\cos(\omega\tau))\omega^2 - 2\alpha\omega\sin(\omega\tau) + \alpha^2}}. \tag{30}$$

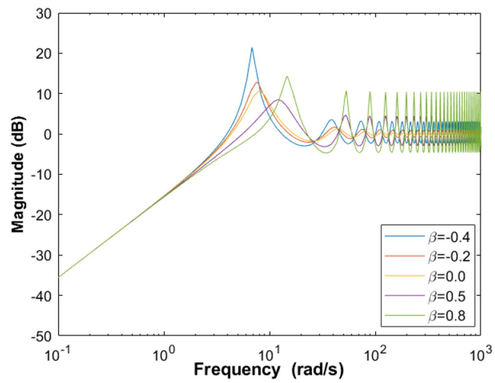

Fig. 6. Frequency response of two-parameter predictor with constant $\alpha = 5$ and $\tau = 0.2$ s, showing the impact of different β values.

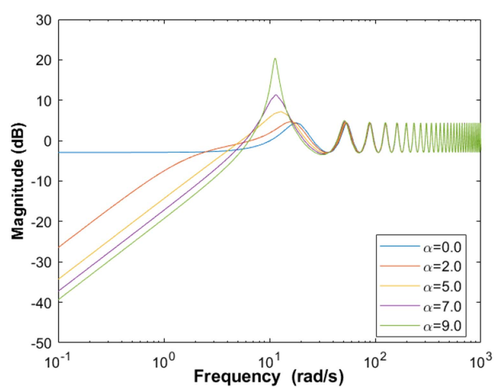

Fig. 7. Frequency response of two-parameter predictor with constant $\beta = 0.5$ and $\tau = 0.2$ s, demonstrating the effect of varying α values.

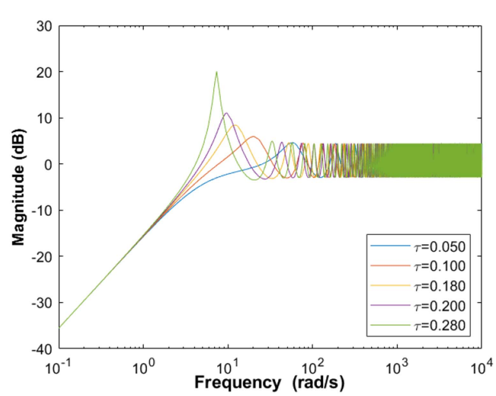

Fig. 8. Frequency response of two-parameter predictor with constant $\beta = 0.5$ and $\alpha = 5$, illustrating the impact of different delay.

A stable predictor does not guarantee effective delay compensation or minimized state error relative to communication-induced coupling error. The frequency response of the two-parameter predictor is analyzed to assess its delay compensation performance. The main objective is to keep the magnitude of (16) below one at relevant frequencies, reducing the prediction state-tracking error relative to the coupling error at a given frequency ω. This helps mitigate the effect of communication delay on the interface coupling signal. Figs. 6–8 present Bode plots for different values of tuning parameters α, β, and delay τ. The first 0 dB point of the error gain indicates the predictor's bandwidth, showing the effective frequency range for delay compensation.

In Fig. 6, with $\alpha = 5$ and $\tau = 0.2$ s, different β designs are analyzed. In the low-frequency range, the error gain remains unaffected by β but decreases as it increases. The bandwidth expands with higher β, though at the cost of reduced high-frequency performance. At high frequencies, the error gain approaches one as frequency increases for $\beta = 0$. For nonzero β, the error gain shows an oscillating response, with oscillation amplitude growing with β. In the medium-frequency range, negative β increases overshoot, while minimal overshoot occurs at a specific positive β. Two main inferences are drawn from the analysis. First, while negative β values are theoretically stable, they reduce predictor performance by increasing overshoot and decreasing bandwidth. Therefore, practical applications should limit β to the positive range between zero and one. Second, a positive β enhances bandwidth and reduces overshoot, suggesting it can improve system performance if the benefits outweigh potential high-frequency oscillations in error gain. This is more suitable for coupling signal systems with low bandwidth.

In Fig. 7, with $\beta = 0.5$ and $\tau = 0.2$ s, the effect of different α values is analyzed. As α increases, the error gain decreases in the low-frequency range. The bandwidth initially drops sharply with rising α before gradually increasing. Higher α leads to greater overshoot and a lower peak frequency in the medium range. The high-frequency response is unaffected, as all curves converge as frequency approaches infinity. These results show that α impacts signal prediction differently from β, confirming that dual tuning parameters enhance design flexibility. In Fig. 8, with $\beta = 0.5$ and $\alpha = 5$, the effects of different delays τ are examined. Increasing τ raises the error gain in the higher low-frequency range and overshoot in the medium-frequency range, while reducing bandwidth. However, τ does not affect the error gain at the lower low-frequency range or the oscillation amplitude at high frequencies. These results indicate that higher delays are harder to compensate for, posing a challenge for delay compensation strategies.

An important inference from Figs. 6–8 is that the optimal tuning of α and β for best performance depends on the delay τ and the coupling signals' frequency. Parameters α and β should be chosen so that the frequency range where the error transfer function magnitude exceeds one lies outside critical frequencies for delay compensation. The frequency-domain analysis shows how α, β, and τ affect the predictor's response. The time-domain behavior is further examined using a typical power system coupling signal at 50 and 60 Hz. Figs. 9 and 10 display the relationship between amplitude distortion and measured delay for one-parameter and two-parameter predictors at these frequencies. In these figures, normalizer (α) and normalizer (α, β) represent the amplitude distortion normalizers for the one-parameter and two-parameter prediction systems, respectively.

In Figs. 9 and 10, the horizontal axis shows the measured delay (τ) in seconds (1 to 10 ms), and the vertical axis shows amplitude distortion. The lines represent amplitude distortion for one-parameter predictors at 50 and 60 Hz. The curves show an increasing trend in amplitude distortion with delay until peaking

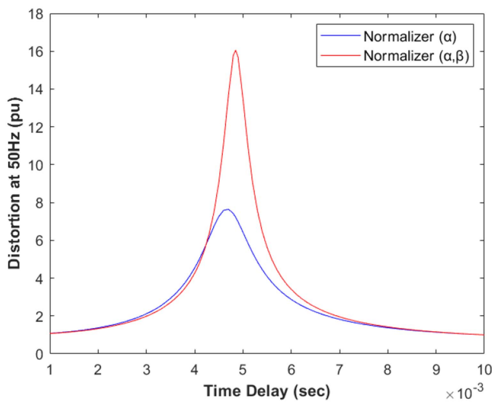

Fig. 9. Coupling signal amplitude distortion for the predicted 50 Hz system signal as a function of measured delay.

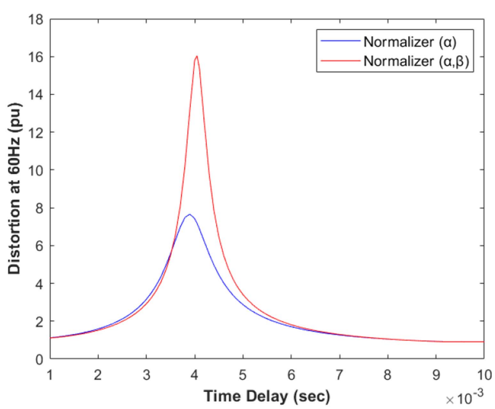

Fig. 10. Coupling signal amplitude distortion for the predicted 60 Hz system signal as a function of measured delay.

and then decreasing. The 60 Hz signals (both predictors) exhibit higher amplitude distortion than the 50 Hz signals, consistent with the frequency domain analysis. Two-parameter predictors generally have slightly higher amplitude distortion than one-parameter predictors at the same frequency. Figs. 9 and 10 show that optimal predictor parameters depend on both delay τ and the coupling signal's frequency, underscoring the importance of frequency-domain analysis for effective delay compensation. This work proposes amplitude distortion normalization, highlighting that amplitude distortion significantly affects predicted signals, even though the phase remains unaffected. Creating a look-up table based on measured delay and distortion factors is essential for signal normalization. Addressing amplitude distortion through normalization, combined with phase delay compensation, enhances model-free delay compensation and maintains low prediction error relative to coupling error at high frequencies.

IV. Experimental Setup and Validation

This study utilized coupling signals from a standard three-phase ITM to validate the model-free delay compensation system, incorporating a proposed look-up table for normalizing signal amplitude. The coupling signals are 60 Hz three-phase voltage and current from a typical power system (transmission and distribution cosimulation) and were exchanged between two real-time simulators situated 45 miles apart at the main and satellite campuses. Fig. 11 illustrates the setup of the two geographically separated laboratories. The real-time cosimulation involved modified IEEE test systems. Subsystem 1, based on the IEEE 5-bus system (transmission), was simulated on Real-Time Simulator 1 in Laboratory 1. This setup extended to Subsystem 2, an IEEE 33-bus system (distribution) connected at load bus 5, simulated on Real-Time Simulator 2 in Laboratory 2. The ITM interface at bus 5 employed controlled current sources, while the substation at bus 1 of the IEEE 33-bus system utilized a controlled voltage source. Three-phase terminal voltages and three-phase currents were exchanged between the subsystems, as shown in Fig. 1.

Data acquisition and exchange between the two laboratories were conducted using Raspberry Pis with VILLASnode software to facilitate coupling signal exchange between the real-time simulators (two Speedgoat Performance Real-Time Target Machines). These simulators house the partitioned subsystems and predictor systems. Accurate real-time communication delay estimation was achieved using the SEL-2488 Satellite-Synchronized Network Clock, with signals timestamped via IEEE 1588 Precision Time Protocol to estimate delays between sent and received signals. The measured delays, Δ_1 and Δ_2, for the respective laboratories, shown in Fig. 12, are inputs to the prediction system and account for time variance in signal transmission. Delays typically range from 4 to 6 ms, with occasional spikes due to bandwidth variations.

The coupling signals $v_{abc1}(t)$ and $i_{abc2}(t)$ are transmitted as line-to-ground $v_{a1}(t)$, $v_{b1}(t)$, $v_{c1}(t)$, and $i_{a2}(t)$, $i_{b2}(t)$, $i_{c2}(t)$ between Subsystems 1 and 2. For model-free prediction, signal derivatives are also exchanged to enable delay compensation. In the cosimulation, Subsystem 1 sends the three-phase voltage signals $v_{abc1}(t)$ to Subsystem 2 and receives predicted three-phase current $i_{abc1}(t)$ using two-parameter prediction as shown in Fig. 13. Similarly, Subsystem 2 transmits the three-phase current $i_{abc2}(t)$ to Subsystem 1 and receives the predicted three-phase voltage $v_{abc2}(t)$ with two-parameter prediction as shown in Fig. 14.

Two robust signal transformation methods, DP and SRF, were tested alongside the single-parameter and two-parameter prediction systems to validate and assess compensation performance. Fig. 15 shows the impact of communication delay and transmission rate on the single-phase signal $v_{a1}(t)$ from the three-phase voltage $v_{abc1}(t)$ and compares the phase compensation capabilities of the prediction systems and traditional methods. $v_{a1}(t)$ is the sent voltage signal, $v_{a2}(t)$ is the received uncompensated signal, $v_{a2}(t)\alpha$ and $v_{a2}(t)\alpha\beta$ are the one- and two-parameter predicted signals, and $v_{a2}(t)$DP and $v_{a2}(t)$SRF are the compensated signals using DP and SRF methods. Fig. 16 illustrates similar effects on current signals, showing how the prediction system with amplitude distortion normalization mitigates phase shifts and distortion. $i_{a2}(t)$ is the sent current signal, $i_{a1}(t)$ is the received uncompensated signal, with $i_{a1}(t)\alpha$, $i_{a1}(t)\alpha\beta$, $i_{a1}(t)$DP, and $i_{a1}(t)$SRF representing their respective compensated signals.

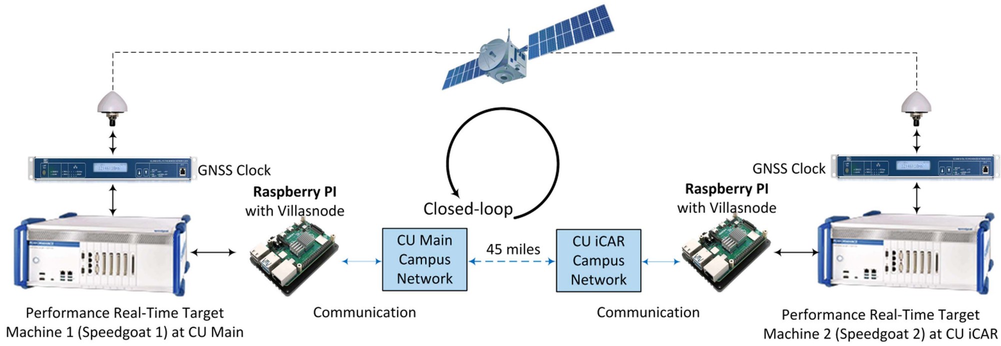

Fig. 11. Schematic of laboratory infrastructure for real-time cosimulation between the main campus and the satellite research campus.

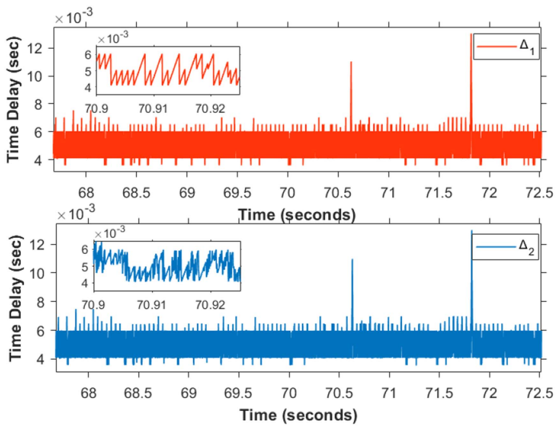

Fig. 12. Estimated communication delays between Subsystems 1 and 2.

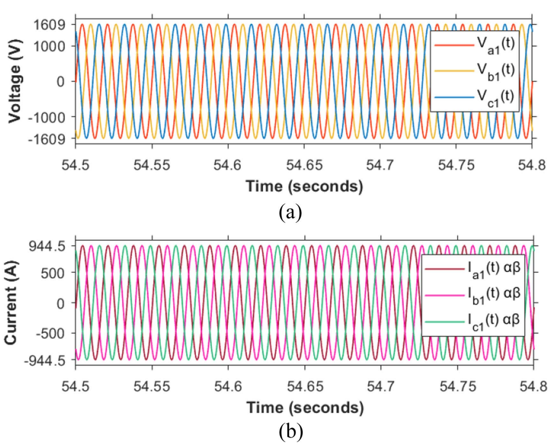

Fig. 13. Coupling signals at ITM Interface 1 (a) sent three-phase voltage signals and (b) predicted three-phase current signals.

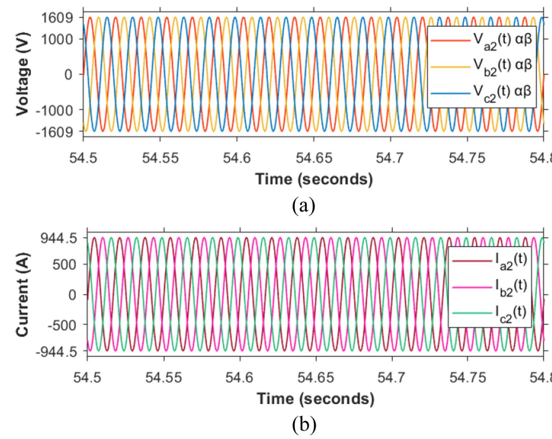

Fig. 14. Coupling signals at ITM Interface 2 (a) predicted three-phase voltage signals and (b) sent three-phase current signals.

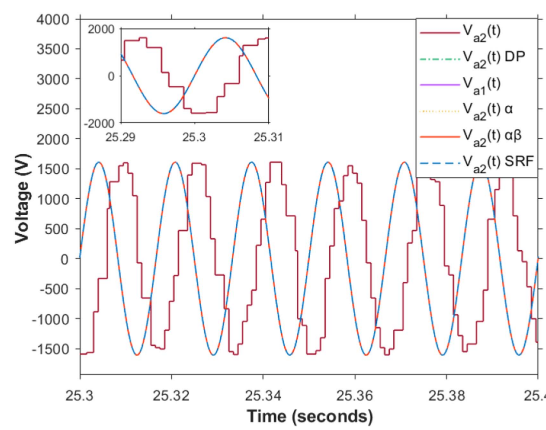

Fig. 15. Impact of communication delay and compensation on single phase $v_{a1}(t)$ of the voltage signal $v_{abc1}(t)$.

To evaluate the performance of the prediction systems (single- and two-parameter), real-time errors between sent and received voltage and current signals were measured. These errors, called coupling errors, result from communication delay and serve as inputs for the prediction systems, as shown in Figs. 17 and 18. The average coupling error values are 2900 V for $v_{a1}(t)$ and 1550 A for $i_{a2}(t)$. The prediction systems aim to minimize these errors. Figs. 19 and 20 separately show the state tracking errors to compare prediction methods with traditional transformation methods. The alternating nature of coupling errors indicates the nature of the coupling signals and highlights the need for effective prediction mechanisms to ensure stability. $e_v(t)\alpha$ and $e_v(t)\alpha\beta$ represent voltage tracking errors for one- and two-parameter methods, while $e_v(t)$DP and $e_v(t)$SRF represent DP and SRF methods. Between 27.514 and 27.5185 s, $e_v(t)\alpha\beta$, $e_v(t)\alpha$, $e_v(t)$SRF, and $e_v(t)$DP peaked at 2.90, 2.92,

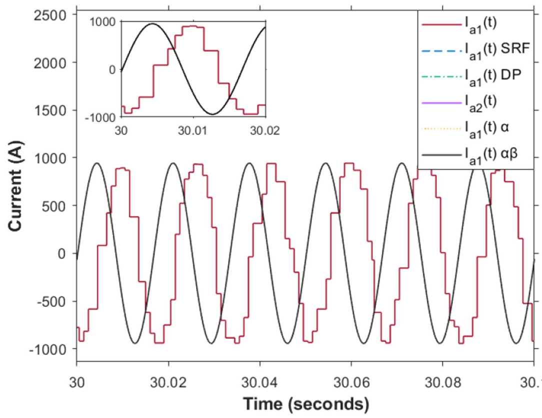

Fig. 16. Impact of communication delay and compensation on single phase $i_{a2}(t)$ of the current signal $i_{abc2}(t)$.

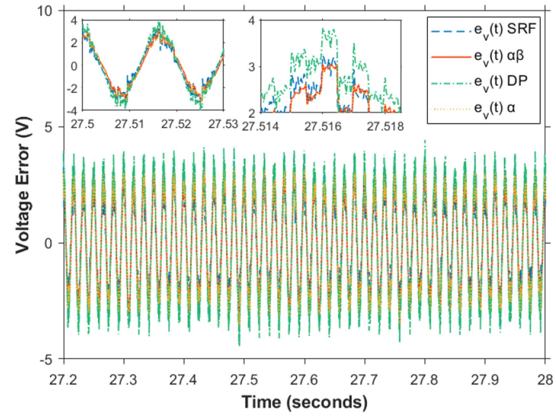

Fig. 19. State errors between sent and received voltage $v_{a1}(t)$.

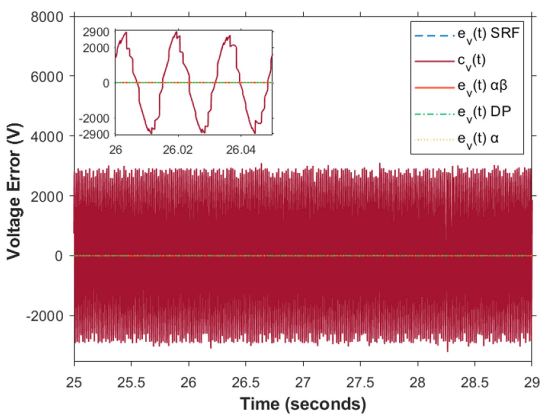

Fig. 17. Coupling and state errors between sent and received voltage $v_{a1}(t)$.

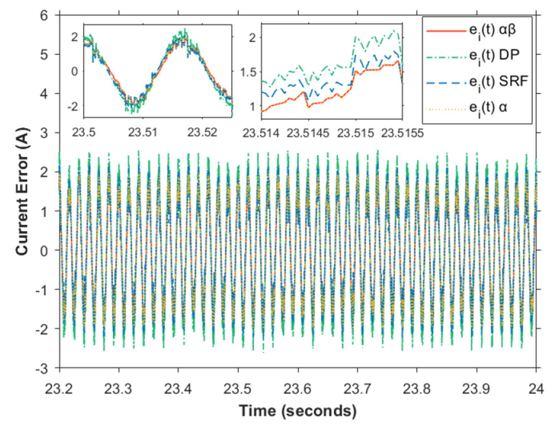

Fig. 20. State errors between sent and received current $i_{a2}(t)$.

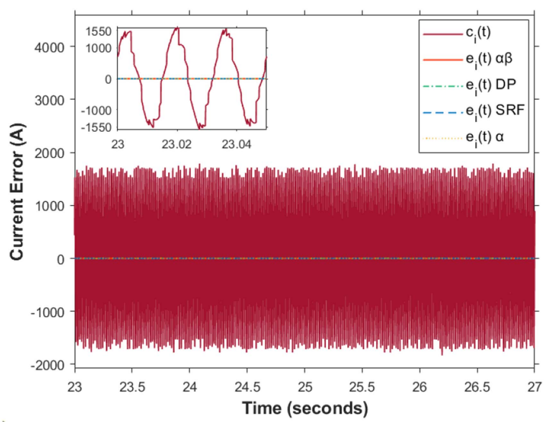

Fig. 18. Coupling and state errors between sent and received current $i_{a2}(t)$.

3.2, and 3.8 V, respectively. Similarly, $e_i(t)\alpha$, $e_i(t)\alpha\beta$, $e_i(t)$DP, and $e_i(t)$SRF denote current tracking errors for these methods. Between 23.514 and 23.5155 s, $e_i(t)\alpha\beta$, $e_i(t)\alpha$, $e_i(t)$SRF, and $e_i(t)$DP peaked at 1.51, 1.52, 1.65, and 2.1 A, respectively. Figs. 19 and 20 show that prediction systems achieve more stable state tracking compared to traditional methods, indicating superior performance. This stability ensures consistent compensation and better handling of signal variations in the coupling signals.

To further assess prediction system performance, residual power—generated at virtual interfaces due to asynchronous computation [20] from communication delays—is analyzed. Calculating three-phase apparent power at each interface provides a comprehensive measure of prediction system effectiveness. Fig. 21 shows discrepancies in complex power at Interfaces 1 and 2 during real-time cosimulation, demonstrating that prediction methods effectively mitigate delay impacts, indicated by reduced residual power. $S_{abc}\alpha\beta$, $S_{abc}\alpha$, S_{abc}SRF, and S_{abc}DP are the power discrepancies in the four methods.

Another fidelity measure is the analysis of frequency discrepancies between Subsystems 1 and 2, shown in Fig. 22, indicating the prediction methods' capability to maintain consistent frequencies across subsystems. $FD\alpha\beta$, $FD\alpha$, $FDSRF$, and $FDDP$ are the frequency discrepancies in the four methods. The ratio of state error to coupling error reflects the prediction system's effectiveness with amplitude normalization. A significant reduction in state error compared to coupling error underscores the system's efficiency in conserving energy and preventing residual energy exchange, maintaining real-time cosimulation fidelity.

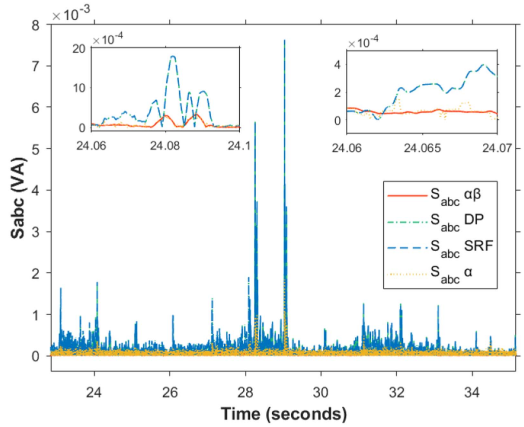

Fig. 21. Discrepancies between the complex power measured at Interfaces 1 and 2.

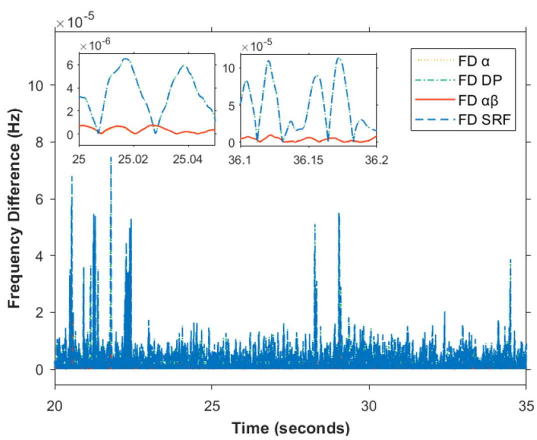

Fig. 22. Frequency discrepancies between Subsystems 1 and 2.

V. Conclusion

In this work, we present a model-free approach for delay prediction and compensation in power system coupling signals for real-time cosimulation, offering an alternative to traditional methods that decompose and reconstruct time-varying signals. This approach aims to reduce inaccuracies and computational complexity. Using sliding mode control, the model-free prediction estimates and compensates for delays within the subsystems' sampling time, ensuring stability, and robustness. Frequency-domain and time-domain analyses were conducted to assess predictor performance and delay compensation capabilities. The frequency-domain analysis shows the impact of tuning parameters and delays on the predictor's response, while the time-domain analysis examines dynamic performance. High-frequency signal studies provided insights into amplitude distortion and signal normalization. The proposed methods were benchmarked against traditional prediction methods, including DP and SRF, demonstrating their effectiveness. Results show advancements in parallel computation and refined coupling point modeling that mitigate latency, maintain system integrity, and improve real-time simulation accuracy.

Acknowledgment

This article was approved for public release and the distribution is unlimited.

References

[1] E. Buraimoh, G. Ozkan, L. Timilsina, P. K. Chamarthi, B. Papari, and C. S. Edrington, "Overview of interface algorithms, interface signals, communication and delay in real-time co-simulation of distributed power systems," *IEEE Access*, vol. 11, pp. 103925–103955, 2023.

[2] M. Manbachi et al., "Real-time co-simulation platform for smart grid volt-VAR optimization using IEC 61850," *IEEE Trans. Ind. Informat.*, vol. 12, no. 4, pp. 1392–1402, Aug. 2016.

[3] A. Khurram, M. Amini, L. A. D. Espinosa, P. D. Hines, and M. R. Almassalkhi, "Real-time grid and DER co-simulation platform for testing large-scale DER coordination schemes," *IEEE Trans. Smart Grid*, vol. 13, no. 6, pp. 4367–4378, Nov. 2022.

[4] P. Palensky, A. A. Van Der Meer, C. D. Lopez, A. Joseph, and K. Pan, "Cosimulation of intelligent power systems: Fundamentals, software architecture, numerics, and coupling," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 34–50, Mar. 2017.

[5] D. Macii and S. Rinaldi, "Time synchronization for smart grids applications: Requirements and uncertainty issues," *IEEE Instrum. Meas. Mag.*, vol. 25, no. 6, pp. 11–18, Sep. 2022.

[6] S. Cheng, C. Li, and K. Zhao, "Research on time synchronization of improved clock servo system based on PTP protocol," in *Proc. IEEE 6th Adv. Inf. Technol., Electron. Automat. Control Conf.*, 2022, pp. 1027–1033.

[7] L. Barbierato et al., "Exploring stability and accuracy limits of distributed real-time power system simulations via system-of-systems cosimulation," *IEEE Syst. J.*, vol. 17, no. 2, pp. 3354–3365, Jun. 2023.

[8] M. Mirz, A. Estebsari, F. Arrigo, E. Bompard, and A. Monti, "Dynamic phasors to enable distributed real-time simulation," in *Proc. 6th Int. Conf. Clean Elect. Power*, 2017, pp. 139–144.

[9] P. G. Papageorgiou, P. T. Papafilippou, K. O. Oureilidis, and G. C. Christoforidis, "An adaptive controller of a hybrid storage system for power smoothing with enlarged battery lifetime," *IEEE Trans. Sustain. Energy*, vol. 15, no. 4, pp. 2567–2580, Oct. 2024.

[10] M. H. Syed et al., "Real-time coupling of geographically distributed research infrastructures: Taxonomy, overview, and real-world smart grid applications," *IEEE Trans. Smart Grid*, vol. 12, no. 2, pp. 1747–1760, Mar. 2021.

[11] R. Liu, M. Mohanpurkar, M. Panwar, R. Hovsapian, A. Srivastava, and S. Suryanarayanan, "Geographically distributed real-time digital simulations using linear prediction," *Int. J. Elect. Power Energy Syst.*, vol. 84, pp. 308–317, 2017.

[12] L. Barbierato et al., "Facilitating smart grids integration through a hybrid multi-model co-simulation framework," *IEEE Access*, vol. 12, pp. 104878–104897, 2024.

[13] E. Bompard et al., "A multi-site real-time co-simulation platform for the testing of control strategies of distributed storage and V2G in distribution networks," in *Proc. 18th Eur. Conf. Power Electron. Appl.*, 2016, pp. 1–9.

[14] M. H. Syed, E. Guillo-Sansano, S. M. Blair, A. Avras, and G. M. Burt, "Synchronous reference frame interface for geographically distributed real-time simulations," *IET Gener., Transmiss. Distrib.*, vol. 14, no. 23, pp. 5428–5438, 2020.

[15] L. Barbierato et al., "Locally distributed digital real-time power system co-simulation via multi-phase distributed transmission line model," *IEEE Trans. Ind. Appl.*, vol. 60, no. 6, pp. 8137–8146, Nov./Dec., 2024.

[16] G. Lauss and K. Strunz, "Multirate partitioning interface for enhanced stability of power hardware-in-the-loop real-time simulation," *IEEE Trans. Ind. Electron.*, vol. 66, no. 1, pp. 595–605, Jan. 2019.

[17] R. Brandl, "Operational range of several interface algorithms for different power hardware-in-the-loop setups," *Energies*, vol. 10, no. 12, 2017, Art. no. 1946.

[18] Y. Zheng, M. J. Brudnak, P. Jayakumar, J. L. Stein, and T. Ersal, "A predictor-based framework for delay compensation in networked closed-loop systems," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 5, pp. 2482–2493, Oct. 2018.

[19] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding Mode Control and Observation*, vol. 10. Berlin, Germany: Springer, 2014.

[20] S. Sadjina and E. Pedersen, "Energy conservation and coupling error reduction in non-iterative co-simulations," *Eng. Comput.*, vol. 36, pp. 1579–1587, 2020.