Improvements to the Co-simulation Interface for Geographically Distributed Real-time Simulation

Steffen Vogel*, Vetrivel Subramaniam Rajkumar†, Ha Thi Nguyen‡, Marija Stevic*,
Rishabh Bhandia†, Kai Heussen‡, Peter Palensky† and Antonello Monti*

*Institute for Automation of Complex Power Systems
RWTH Aachen University
Aachen, Germany
{stvogel, mstevic, amonti}@eonerc.rwth-aachen.de

†Intelligent Electrical Power Grids
Delft University of Technology
Delft, The Netherlands
{v.subramaniamrajkumar}@student.tudelft.nl
{r.bhandia, p.palensky}@tudelft.nl

‡ Department of Electrical Engineering
Technical University of Denmark
2800 Kgs Lyngby, Denmark
{thangu, kh}@elektro.dtu.dk

***Abstract*—As future power systems become increasingly complex and interconnected to other energy carriers, a single research infrastructure can rarely provide the required testbeds to study a complete energy system. Therefore, virtual interconnection of laboratories for large-scale systems plays an important role for Geographically Distributed Real-time Simulation. This paper presents the improvements made in simulation fidelity as well as usability for establishing future simulator and laboratory connections. A general procedure is proposed and analyzed for geographically distributed simulation, which allows users easily to adapt this system to specific test cases. A systematic and comprehensive analysis of dynamic phasor based co-simulation interface algorithm and its improvements are provided to demonstrate the advantages as well as limitations of this approach...**

***Index Terms*—Co-simulation, Dynamic phasors, Geographically distributed real-time simulation, Laboratory coupling, Real-time digital simulator.**

I. Introduction

The complexity of the present and future power system requires tools that are suited for simulations on a large scale. This is in order to study the interactions with newer energy sources and interoperability of new control methods. Advancement in power electronics, driven by the energy transition calls for high-fidelity simulation tools and a shift from static load flow simulations to dynamic ones. The stability of future power system operation with inverter dominated dynamics requires a matching assessment infrastructure.

Real-time power system simulation and testing for geographically dispersed locations is an advanced concept which enhances testing capabilities. Real-time simulation resources, Power Hardware in Loop (PHiL) setups and hybrid co-simulation frameworks may be interconnected to form a comprehensive research infrastructure that allows the sharing of resources and integration of facilities with different hardware setups located far from each other. Thus, virtual interconnection of laboratories will allow for expanding capabilities of individual laboratories to address the requirements for studies of large-scale, system level and interdisciplinary scenarios.

Virtual interconnection of laboratories is a novel concept that has attracted lot of attention of researchers. Its main application is for Geographically Distributed Real Time Simulation (GD-RTS), which addresses the requirement of large-scale simulation resources using real-time simulation at multiple labs for joint simulations [1]–[5]. Additionally, its framework allows the participants to be flexible and confidential for joint experiments based on indirect data sharing. A co-simulation of multi-area power systems was investigated in [6], which dealt with the requirement of large-scale simulation resources using real-time simulation at multiple institutions and the confidence for their grid models. In [7], a GD-RTS of HVDC systems were implemented using ideal transformer model interface algorithm. This work showed that simulation fidelity is robust with large time delay. However, co-simulation interface algorithms should be improved to guarantee simulation fidelity of all quantities.

Previous work [4], [8] has already demonstrated the use of Dynamic Phasors as a feasible approach to couple real-time simulators. This paper presents the improvements made in simulation fidelity as well as usability for establishing future simulator and laboratory connections. A general procedure is proposed and analyzed for geographically distributed simulation, which allows users to easily adapt this system to specific test cases. A systematic and comprehensive analysis of DP-based co-simulation interface algorithm and its improvements are provided to demonstrate the advantages as well as limitations of this approach. The analysis includes multiple

The authors would like to thank the the European Unions Horizon 2020 research and innovation programme projects ERIGrid (GA 654113) and RESERVE (GA 727481) for the support of this work.

transients within a power system that are simulated, as well as characteristics of a communication setup between the laboratories. As part of the H2020 ERIGrid Transnational Access (TA) research exchange, the real-time simulation laboratories at RWTH Aachen and TU Delft have been interconnected for GD-RTS.

The paper is organized as follows: Section II presents the architecture of the experiment setup. The detail communication and simulation test cases are described in Section III. Section IV investigates in the co-simulation interface algorithm based on dynamic phasors. Section V focuses on implementation details and improvements made in comparison to previous versions. Results of the communication test-cases as well as of the simulation scenarios are presented in Sections VI. Some important conclusions are drawn, along with future work to be undertaken in Section VII.

II. Architecture

This section describes the system architecture used for GD-RTS as illustrated in Figure 1.

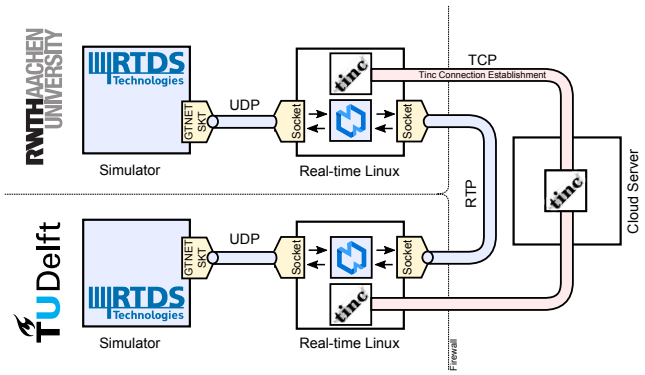


Fig. 1. Architecture

A. Communication Network

Interface signals from the electrical domain are exchanged over the public internet between both Real-time Digital Simulators (RTDS). For best results, both laboratories are connected via their university networks to the national research and education networks (NRENs). Namely, Germany's *Deutsches Forschungs Netzwerk (DFN)* and the Netherlands' *SURFnet* are not-for-profit organizations which are itself interconnected the the European GÉANT network. In contrast to special purpose computing networks such as the LHC Computing Grid (LGC) by CERN [9], the networks used in this paper are part of the public internet and thus, shared with other users, which may result in unpredictable spikes in communication latency or packet loss. In this paper, we demonstrate how the Dynamic Phasor Interface Algorithm described in section IV can mitigate the consequences of these disturbances.

1) Virtual Private Network: Tinc: For GD-RTS it is crucial to establish a direct point-to-point connection between the labs to maintain low latency and jitter. As in most cases, computing equipment in both laboratories is protected against external threats via a firewall which filters incoming network traffic. There are two approaches to circumvent this restriction:

1) Virtual Private Networks (VPN) can tunnel network traffic between both sites. Most of the time this requires the addition of exceptions to the firewall rule-set.
2) A technique referred to as *Interactive Connectivity Establishment* which facilitates peer-to-peer networking as used by Voice-over-IP (VoIP) or other Video Conferencing Solutions such as Microsoft's Skype [10].

In this work Tinc-VPN software has been used which leverages both above mentioned techniques to establish a fully meshed VPN network between multiple labs [11]. It does so by using a publicly reachable node which supports the restricted nodes in their connection establishment. However, after this initial connection establishment, all critical real-time traffic is exchanged peer-to-peer between the laboratories.

B. Co-simulation Gateway: VILLASnode

Within each laboratory a Linux-based gateway server is deployed to establish VPN connections and handle data exchange between the local simulators and remote laboratories. For the data exchange *VILLASnode*, a component of the *VILLASframework*, is used [12]. VILLASnode is a C/C++ application tuned for the real-time exchange of simulation data in different formats and protocols. In this setup, it handles the collection of statistics on the communication link as well as the protocol conversion between the UDP-based connection to RTDS' GTNET card and the Real-time Transfer Protocol (RTP) which has been used between the laboratories.

C. Real-time Protocol: RTP / RTCP

The frequency with which simulators exchange their interface signals is one of the factors which affects accuracy of simulation results. The maximum rate is limited by the available bandwidth of the communication links. As the link is shared with other users, the available bandwidth varies with time, which may cause congestion. Congestion has to be avoided as it leads to packet loss and a degradation of simulation fidelity.

In the past, a static rate has been determined by empirical tests preceding the actual simulation runs. This approach is cumbersome as it needs to be repeated for every new communication link and possibly at different times of a day. To improve this situation, this paper implements a congestion avoidance scheme based on a real-time protocol which dynamically adjusts the sending rate.

For data exchange between the gateway nodes, the RTP and its sibling the Real-time Control Protocol (RTCP) are used [13]. Both protocols are used in conjunction: RTP handles data transfer whereas RTCP is used to exchange Quality of Service (QoS) reports which measure packet loss, delay and jitter. These reports are the main advantage of RTP over plain User Datagram Protocol (UDP) packets, as they can be used by the sender to adjust its behaviour.

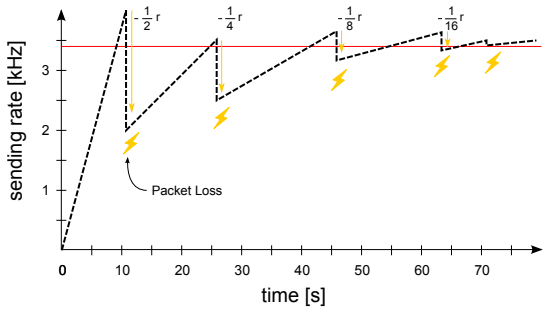


Fig. 2. AIMD sending rate adaption with decreasing impact over time.

RTP and RTCP are widely used protocols in real-time multimedia streaming applications such as Voice-over-IP or Video-on-Demand streaming. In these applications, RTCP receiver reports are used to adjust the bit rate of multimedia codecs based on the current network conditions in order to avoid stuttering and lags the stream.

In the context of this work, the receiver reports are used to change the sampling (sending) rate of signals which are a exchanged between the simulators as shown in Figure 2. The congestion avoidance algorithm is implemented in the VILLASnode gateway. It is inspired by TCP Additive Increase, Multiplicative Decrease (AIMD) scheme which in the case of no congestion, increases the sending rate linearly and in case of congestion, reduces the sending rate by a multiplicative factor [14]. In order to avoid large changes in the sending rate which could affect the simulation results, the impact of the AIMD adaptation is reduced by half with every occurring congestion event.

RTP encapsulates the simulation signals in a raw IEEE 754 single-precision floating format and adds header fields for time-stamps as well as a sequence number. In this test, RTP uses UDP as the underlying transport protocol.

D. Real-Time Digital Simulators: RTDS

Both labs operate digital real-time simulators from RTDS Technologies. During this experiment, two Novacor chassis at RWTH and one PB5 rack at TU Delft have been used to simulate the system described in Section III. For synchronization, both RTDS installations are equipped with GTSYNC extensions cards which use the Global Position System (GPS) to synchronize the time-steps as well as the simulation start to a common time reference. However, as seen later, synchronization is not a essential requirement for the fidelity of the simulation. It is instrumental however, for the collection and alignment of results.

One important contribution of this work is the development and release of a reusable library component and its demonstration in a demo model for the RTDS RSCAD software. It simplifies the procedure to adapt existing models for a GD-RTS as the user can simply copy a library block into his/her model.

III. Test Cases

A. Communication Tests

In order to interconnect the real time simulators at both labs, the network topology and underlying characteristics of the communication network have to be studied. The following communication tests are conducted, for the same:

1) Ping & Trace-route tests: The most basic of communication tests, involves using the standard ping and trace-route tools to gain a basic idea about the network connecting the two sites. For the ping test, 10000 packets in total, in intervals of 10 ms are transmitted from one VILLASnode gateway machine to the other and average RTT, recorded. This is found to be in the range of 12.7 to 13 ms. Furthermore, using the trace-route tool, the routing path between RWTH Aachen and TU Delft is found to be as depicted in figure 3. The total number of hops is 17. Using the mtr (My Traceroute) command, each individual hop is pinged with 1000 packets in intervals of 100 ms and the resulting statistics of the network can be stored. This is discussed in greater detail in section VI.

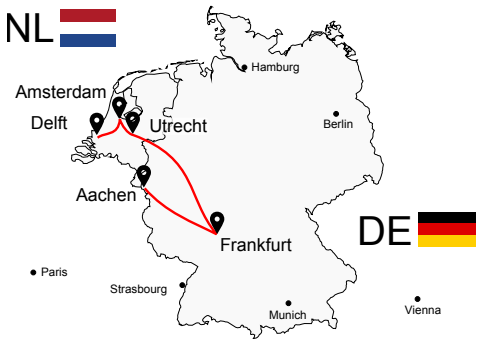


Fig. 3. Reconstructed routing path between Aachen and Delft.

2) Between VILLASnode Gateways: In this test, data is exchanged between the VILLASnode gateway machines running Tinc VPN in a *loop-back configuration* through UDP. A sine-wave with certain frequency is generated at either location and every sample is transmitted to the other site, with a time-stamp. The received samples on the other end, are re-transmitted back to the source without any changes to the encapsulated data. By comparing the sent and received time-stamps, the communication latency between the sites is estimated. This test is carried out for the following combinations:

- With a fixed sending rate and variable data size
- With a variable sending rate and fixed data size

3) Between Simulators using GTSYNC: As the final step of communication testing, data exchange between both the real-time simulators is carried out. To achieve this, models which test the communication latency, jitter and maximum packet rate were developed. Using the GTSYNC cards at both labs, simulations can be started concurrently.

Simulation data is transferred from one RTDS to its local VILLASnode machine through the GTNET card and GTNET SKT (socket) protocol The SKT protocol represents both

integer and floating point numbers as 4 bytes in the packet. Floating point numbers are single precision(32-bit) and are represented using IEEE754 format. The packet size is an integral multiple of 4 bytes, depending on number of data points sent. From there on, this data is sent through the internet to the other lab, where is it received by its VILLASnode machine and transferred to the RTDS using a similar setup. The IP addresses, ports and variables to be exchanged are specified in the RSCAD model files.

B. Simulation Test Cases

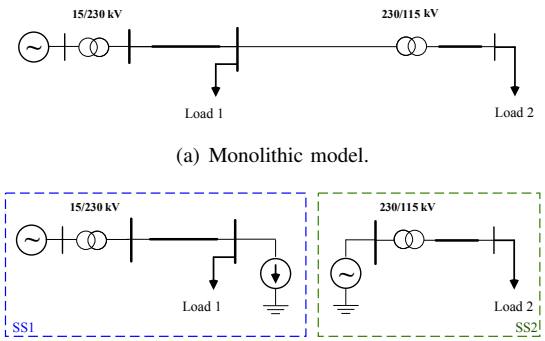
The system under study in this paper is based on a simple power system consisting of a voltage source connected to loads through a transmission line as shown in Figure 4. This system represents the most simple version of a transmission / distribution network, where the co-simulation interface is located at the transfer point. Initial tests have been conducted with an ideal voltage source to validate the co-simulation setup. As the ideal voltage source leads to a constant system frequency, it is not suitable for analysing frequency transients and their propagation across the co-simulation interface. Hence, for subsequent tests, the ideal voltage source has been replaced by a synchronous generator.

For validation, three different variants of the model have been compared:

1) Monolithic model: Monolithic model is modelled in real-time simulator using a single RTDS rack where the entire system is simulated in a single subsystem with 50 µs time step as shown in Figure 4(a). It serves as a reference with no time delay for the comparison of the decoupled and distributed model.

2) Decoupled model: The decoupled model shown in Figure 4(b) is derived from the monolithic model where the system is separated into two subsystems at the 230 kV bus based on Ideal Transformer Model (ITM) approach for co-simulation as described in Section IV. The simulation is executed across two RTDS racks but still contained and started by a single RSCAD/draft file. Signal exchange between the subsystems is handled by cross-rack signal import/exports. Additional communication delay can optionally be applied to these cross rack signals by using RSCAD control components.

3) Distributed model: The distributed model is derived from the decoupled model by moving one subsystem entirely to a separate RTDS simulation. The co-simulation now spans two independent RSCAD/draft files which are started separately. The signal exchange of the subsystems is handled by the VILLASnode gateway described in Section II-B.



(a) Monolithic model.

(b) Decoupled/distributed model.

Fig. 4. Simple power system diagram.

IV. Co-simulation Interface Algorithm based on Dynamic Phasors

Co-simulation interface algorithm is based ITM approach. As illustrated in Figure 4(b), controlled sources are utilized to impose voltage and current measured at the interface. DRTS systems perform transient simulation in time domain where current and voltage quantities are represented with its wave forms. Direct sampling and transfer of instantaneous values of the wave forms across a shared wide-area communication network that is characterized by relatively large and time-varying delay would significantly deteriorate the wave forms and simulation fidelity. To this aim, two main approaches are proposed in the literature. Representation of interface quantities in the form of Root Mean Square (RMS), frequency and phase angle is proposed in [15]. In this work, co-simulation interface algorithm based on representation of current and voltage at the interface in the form of time-varying Fourier coefficients [4], known as Dynamic Phasors (DP), is utilized.

Figure 5 illustrates transformation of wave forms to a DP-based representation before sampling and sending interface quantities to the remote DRTS system. At the receiving side, the DP quantities are transformed back to the time-domain waveform to provide reference for the controlled sources as illustrated in the Figure 6. DP interface algorithm allows for compensation of time-varying delay based on phase shift within reconstruction of time-domain waveform. This approach has been proposed for Power Hardware-In-the-Loop (PHIL) in [16]. Details of the implementation and comprehensive analysis of the DP-based co-simulation interface algorithm are given in the following Section V.

V. Implementation Details

A. DFT Window Length

In a DRTS, the power system is simulated in the time domain through discrete and equidistant time steps. As a consequence, the Fourier transform used to calculate dynamic phasors must be a Discrete Fourier Transform (DFT). The DFT is implemented as a series over the moving window of the product of signal $x[n]$ with reference phasors:

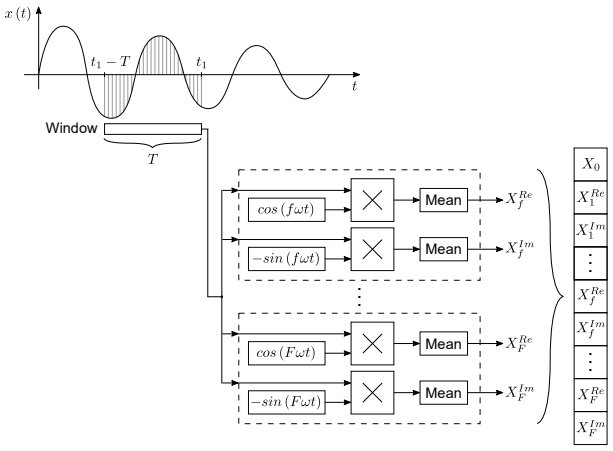


Fig. 5. Calculation of DP coefficients

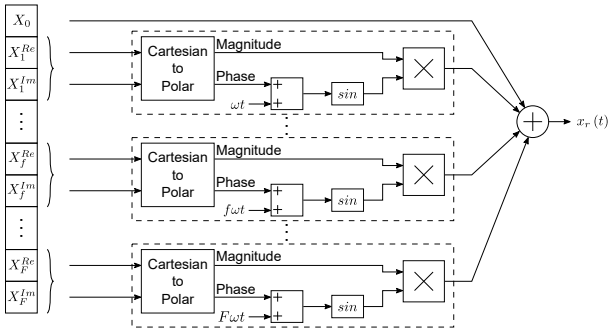


Fig. 6. Reconstruction of the time-domain waveform based on DP coefficients

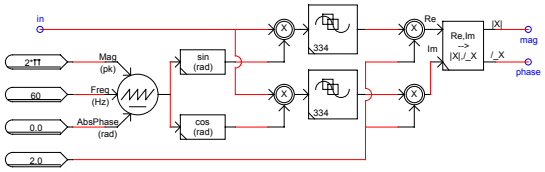
$$X_k[m] = \frac{1}{N} \sum_{n=m-N}^{m} x[n] \cdot e^{-j2\pi f_0 kn} \tag{1}$$

As a result, the Fourier coefficients describe the harmonic components of the interface quantity and the system. The length of the DFT window is chosen so that it covers one period of the fundamental frequency of the signal. For example, in a 60 Hz system which is simulated with a 50 µs time step T_d, the window has a length of $N \approx (1/f_0)/T_s) = 333,\overline{3}$. As this window length is not an integral number, the resulting Fourier coefficients do not represent the harmonic components of the 60 Hz system accurately. If this error is not properly taken into account, RMS values of interface quantities will not be identical and the power exchanged at the co-simulation interface will be imbalanced even in steady state. As indicated in the Table I, a possible solution to this issue is the adjustment of the simulation time step such that, the DFT window length is closer to an integral number: $T_s = (1/f_0)/334 \approx 49,9$ µs

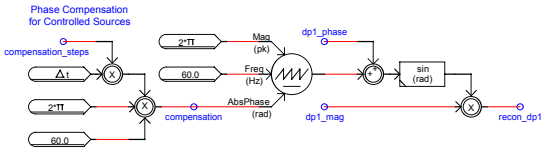
It is to be noted that, this problem only occurs for systems whose fundamental period is not evenly divisible by the simulation time step. E.g. simulations with a 50 Hz system frequency and a 50 µs time step are more accurate as the DFT window exactly spans 400 steps.

TABLE I
RMS VALUES OF INTERFACE QUANTITIES WITH RESPECT TO THE DFT WINDOW LENGTHS.

DFT window			Interface quantity			
			$V_{A,rms}$ [kV]		$I_{A,rms}$ [A]	
T_s [µs]	N	$length$ [ms]	SS1	SS2	SS1	SS2
50	333	16.65	136.7	136.0	51.64	51.9
50	334	16.7	136.6	137.9	52.56	52.05
49.9	334	16.6666	136.6	136.6	52.56	52.56



(a) DFT with moving average blocks.



(b) Signal reconstruction.

Fig. 7. RSCAD implementation of DP-IA for a single frequency component.

B. DFT Calculation

A critical step for performance of the Dynamic Phasor Interface Algorithm (DP-IA) is the calculation of complex-valued phasors from the real-value instantaneous voltage and current signals and their subsequent reconstruction to original form. Phasors are calculated by a DFT from which only certain harmonic components are selected. The calculation is continuously updated over a moving window, thereby producing a stream of *dynamic* phasor updates. Different approaches to calculate the phasors and the reconstruction have been considered:

1) RSCAD: DFT control component: The included DFT block in RSCAD supports a maximum sampling of 64 points per cycle. For the case of a 60 Hz system, this results in a maximum update rate of the phasors to $60\ \text{Hz} \times 64 = 3840$ Hz. The execution time required for this block is $0.18\ \mu\text{s} \times 64 + 0.5\ \mu\text{s} = 12,02\ \mu\text{s}$ which is relatively high. In a standard co-simulation, 9 DFT blocks are required (3 phases $\times$ 3 harmonic components). Given the execution time, this would result in a utilization of several control component processors which is unacceptable.

2) RSCAD: Moving average window: Figure 7(a) shows a custom implementation of the phasor calculation which utilizes moving average blocks in RSCAD. This implementation produces an updated phasor in every simulation time step and has an execution time which is nearly half of that of the DFT block.

3) VILLASnode: Both RSCAD based DFT implementations suffer from the disadvantage that, changing the number of harmonic components is not feasible without extensive manual changes to the models and their large utilization of RTDS

resources. Therefore, it is desirable to move the calculation of the DP interface algorithm to the VILLASnode gateway and therefore reduce the amount of changes required to the initial model. The only modification necessary to adapt a model for co-simulation is be the addition of communication blocks (GTNET) and controlled sources. The simulation gateway has been updated to perform calculation and reconstruction of phasors to instantaneous values.

C. Update Rate

In steady state, all three approaches provide correct results as the exchanged phasors remain constant.

During transients the update rate of the phasors becomes critical. There are currently three bottlenecks which limit the update rate of the phasors: First, the RSCAD DFT block is limited to 3840 Hz. Secondly, RTDS specifies the maximum sending rate of its GTNET card with 5 kHz. At last, the VILLASnode gateway might restrict the sending rate in order to avoid congestion on the communication link as described in Section II-C.

The first bottleneck has been solved by using the custom DFT implementation based on moving average windows. The second bottleneck is the current limiting factor and can be solved in the future by replacing the GTNET with with a more potent GTFPGA card. By utilizing the PCIexpress and Aurora interfaces of an FPGA board, a signal exchange between the gateway and the simulator in every time step becomes possible. Therefore the simulations in this paper have been conducted with a update rate of 5 kHz.

In any case, a sending rate less than the simulation time step ($f_c < \frac{1}{T_s}$) during transients will result in discontinuities in the reconstructed signal. In order to be avoid system instabilities, the signal should be filtered with a low pass filter. Initial tests with a 3rd degree Butterworth filter with $f_c = \frac{1}{2T_s}$ have shown promising results but entail a group delay of the reconstructed signal. This delay will be added on top of the existing communication delay.

D. Signal reconstruction of time-domain waveform

The reconstruction of the instantaneous voltage/current wave forms is performed by multiplying the dynamic phasor coefficients $X_k[n]$ with the same rotating reference phasor as in V-B. The resulting signal is then used to directly control a voltage/current source at the coupling point. Due to internals of RTDS scheduling, the resulting reconstructed waveform is always delayed by 1-3 time steps n_c as the control components cannot control the sources in the same time step as they are executed. This demands for a constant phase compensation by $\varphi_c = 2\pi f_0 n_c T_s$ which can be added to the absolute phase of the reference phasor as shown in Figure 7(b):

$$x[n] = \sum_{k=0}^{K} X_k[n] \cdot e^{j(2\pi f_0 k n + \varphi_c)} \tag{2}$$

Table II shows the steady state power flow with respect to different compensation steps for the voltage and current sources in SS1 and SS2 respectively. Note that the apparent power and RMS of the voltage at the interface are matching in all cases. If internal delays of controlled sources are properly compensated, the power flow is identical for both sides of the interface ($P_{SS1} = P_{SS2}, Q_{SS1} = Q_{SS1}$) which is given for $n_{SS1} = 3, n_{SS2} = 2$.

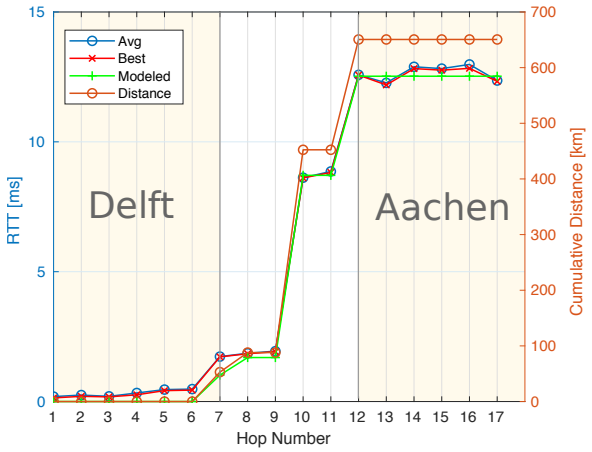


Fig. 8. Latency of Routing Path

TABLE II
STEADY-STATE POWER FLOW WITH RESPECT TO PHASE COMPENSATION OF SOURCE SIGNALS.

n_{SS1} [T_s]	n_{SS2} [T_s]	P_{SS1} [MW]	Q_{SS1} [MVar]	P_{SS2} [MW]	Q_{SS2} [MVar]	S [MVA]	$Vrms$ [kV]
0	0	19.16	9.846	20.0	8.003	21.54	227.7
1	1	19.52	9.118	20.0	8.003	21.54	227.9
2	1	19.69	8.749	20.0	8.003	21.54	227.9
3	2	20.0	8.003	20.0	8.003	21.54	228.1

VI. Results

A. Communication Tests

The communication results between the VILLAS node gateways and the real-time simulators are presented here. Figure 8 shows the statistical analysis of all the hops on the routing path. It can be observed that there is a positive correlation between the actual physical distance between hops and the modeled latency, calculated using $T_m = a * D * (c/n)$ with $a \approx 2$ as an air-line distance correction, D as the distance, c as speed of light and $n \approx 1,5$ as the refractive index of a fiber optic.

To gain greater insight into the round trip times, a detailed statistical analysis is performed. Figure 9 shows the cumulative distribution of Round Trip Times (RTT) for the transfer of a fixed number of data points–24 with variable sending rates between the gateway machines. 24 data points are chosen as they are relevant to the application/use-case. It can be inferred from figure 10 that, for sending rates greater than 10000 packets/sec, RTT is higher than average and they can be treated as outliers. Lower sending rates however, are tightly coupled.

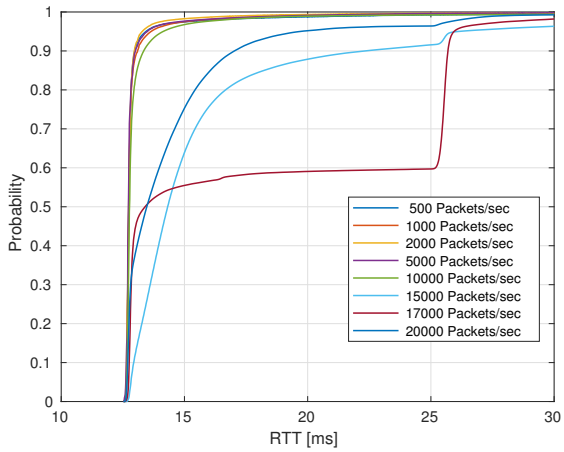


Fig. 9. Cumulative distribution of RTT for varying sending rates

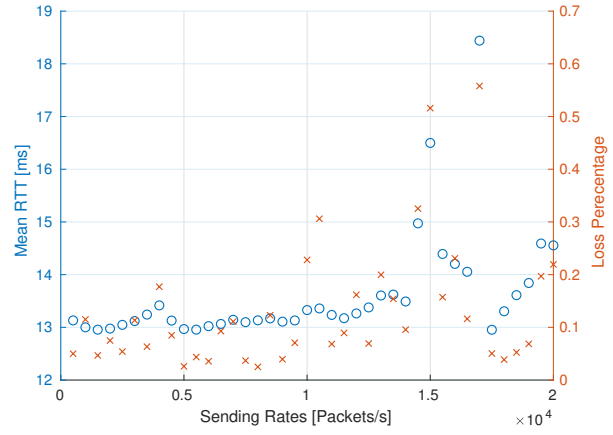


Fig. 10. RTT statistics for variable sending rates with fixed data point

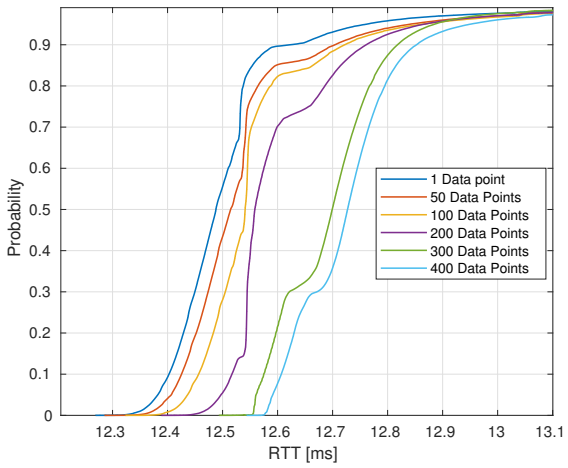


Fig. 11. Cumulative distribution of RTT for varying data sizes

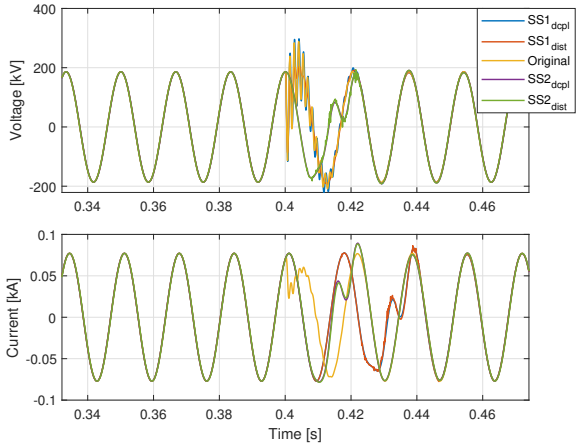


Fig. 12. Limits of DP-IA interface in case of 90 deg phase jump.

For instance, with a sending rate of 17000 packets/sec, there is only a 60% chance of the RTT being lesser or equal to 25 ms.

A similar analysis is carried out for RTT with a fixed sending rate of 2000 packets/sec with variable data points. This is seen through Fig. 11. Thus, it can be concluded that, neither higher sending rates nor large data sizes are preferable. This can be attributed to the fact that, the communication network between the two sites is not a dedicated one, but shared. Hence, this introduces a degree of stochasticity or unpredictability.

B. Co-Simulation

1) Limits of the co-simulation interface: Figure 12 shows the propagation of a 90° phase discontinuity of the ideal voltage source. It shows that during this fault like event, the interface quantities strongly differ from the monolithic reference signal. In the case of this simple and stiff power system which is used in this publication, the system quickly returns to steady state.

2) Effects of Jitter and Congestion on Interface Fidelity: Figure 13 shows variations in the system frequency at the interface point which are correlated with spikes in the communication latency.

C. Validation of the Co-simulation Interface

Figures 14 to 16 show instantaneous, RMS and power quantities on both sides of the interface. If compared against the original monolithic model as well as the decoupled version, only a slight delay of 6 ms is observable.

VII. Conclusion and Future Work

This paper identified and addressed several issues in the co-simulation interface which has been used so far for GD-RTS.

Significant differences in Quality of Service of the communication link have been observed in comparison to previous lab couplings. These demand for an automated approach to monitor the network and to adapt to network congestion. The paper introduces the application of RTP/RTCP protocols in the real-time simulation domain.

In the current state, GTNET cards cannot achieve the required sending rate which is necessary to move the inter-

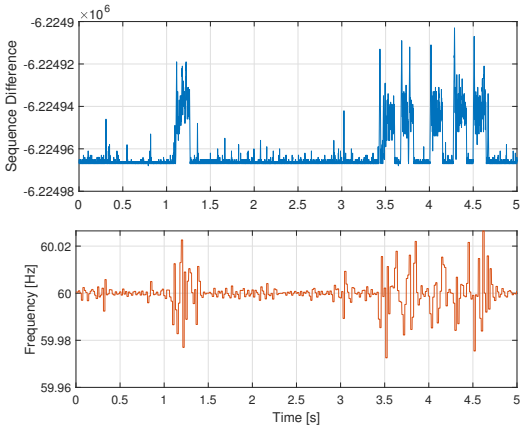


Fig. 13. Frequency variations at the interface caused by spikes in communication latency.

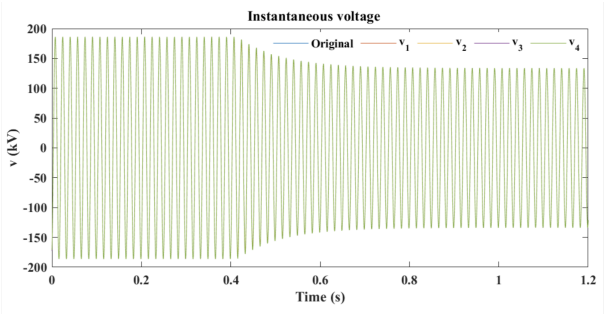


Fig. 14. Instantaneous signals at interface for change of source amplitude.

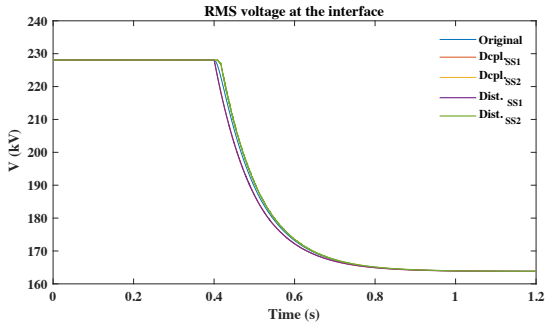


Fig. 15. RMS Voltage at interface for change of source amplitude.

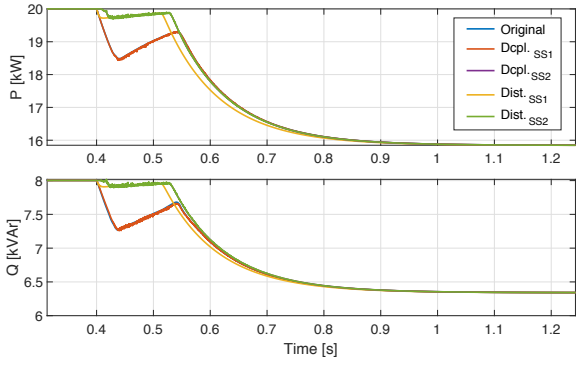


Fig. 16. Active / Reactive power flow at interface for change of source amplitude.

face algorithm into the VILLASnode gateway. In the future, GTFPGA cards could be used to work around this issue.

Two sources of error have been also identified: Firstly, DFT window lengths do not always match with the fundamental system period due to the usage of discrete time steps by the simulators. This error has been successfully reduced by adjusting the simulation time step to a integral factor of the period. Secondly, constant phase offset caused by scheduling dependencies in the control system and network solution within the RTDS system have been identified and solved by applying a constant compensation to the phase of the reconstructed signal.

Last but not least, a re-usable library block for RSCAD has been developed and tested [17]. RTP/RTCP support as well as an implementation of the interface algorithm has been implemented for VILLASnode. Both contributions are released under an open-source licence to enable other researchers to easily adapt new models for GD-RTS and to setup future lab connections.

Based on the simple power system scenario, a complex system, composed of a transmission (IEEE 9-bus system) and a distribution system (IEEE 34-node test feeder), is planned to be implemented among three real-time simulation laboratories at RWTH Aachen university, Technical University of Denmark (DTU) and Delft University of Technology (TU Delft) for further testing and research. Additionally, there is an experimental facility, known as SYSLAB, designed as a testbed for advanced control and communication concepts for power grids. One of a feeder of SYSLAB will be used for power hardware-in-the-loop for the co-simulation, which is named multi-rate setups...

References

[1] M. Stifter, J. H. Kazmi, F. Andrn, and T. Strasser, "Co-simulation of power systems, communication and controls," in *2014 Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, April 2014, pp. 1–6.

[2] Hua Lin, S. Sambamoorthy, S. Shukla, J. Thorp, and L. Mili, "Power system and communication network co-simulation for smart grid applications," in *ISGT 2011*, Jan 2011, pp. 1–6.

[3] R. Bottura, D. Babazadeh, K. Zhu, A. Borghetti, L. Nordstrm, and C. A. Nucci, "Sitl and hla co-simulation platforms: Tools for analysis of the integrated ict and electric power system," in *Eurocon 2013*, July 2013, pp. 918–925.

[4] M. Stevic, A. Estebsari, S. Vogel, E. Pons, E. Bompard, M. Masera, and A. Monti, "Multi-site european framework for real-time co-simulation of power systems," *IET Generation, Transmission Distribution*, vol. 11, no. 17, pp. 4126–4135, 2017.

[5] C. Steinbrink, F. Schlgl, D. Babazadeh, S. Lehnhoff, S. Rohjans, and A. Narayan, "Future perspectives of co-simulation in the smart grid domain," in *2018 IEEE International Energy Conference (ENERGYCON)*, June 2018, pp. 1–6.

[6] TUDelft. Power systems co-simulation. [Online]. Available: https://www.tudelft.nl/en/eemcs/the-faculty/departments/electrical-sustainable-energy/intelligent-electrical-power-grids-iepg-group/projects/current-projects/co-simulation/

[7] D. Shu, X. Xie, V. Dinavahi, C. Zhang, X. Ye, and Q. Jiang, "Dynamic phasor based interface model for emt and transient stability hybrid simulations," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 3930–3939, July 2018.

[8] M. Stevic, A. Monti, and A. Benigni, "Development of a simulator-to-simulator interface for geographically distributed simulation of power systems in real time," in *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, Nov 2015, pp. 5020–5025.

[9] C. Eck, J. Knobloch, L. Robertson, I. Bird, K. Bos, N. Brook, D. Dllmann, I. Fisk, D. Foster, B. Gibbard, C. Grandi, F. Grey, J. Harvey, A. Heiss, F. Hemmer, S. Jarp, R. Jones, D. Kelsey, M. Lamanna, H. Marten, P. Mato-Vila, F. Ould-Saada, B. Panzer-Steindel, L. Perini, Y. Schutz, U. Schwickerath, J. Shiers, and T. Wenaus, *LHC computing Grid: Technical Design Report. Version 1.06 (20 Jun 2005)*, ser. Technical Design Report LCG. CERN, 2005. [Online]. Available: http://cds.cern.ch/record/840543

[10] J. Rosenberg and C. Holmberg. Interactive connectivity establishment (ICE): A protocol for network address translator (NAT) traversal. [Online]. Available: https://tools.ietf.org/html/rfc8445

[11] G. Sliepen, "The difficulties of a peer-to-peer VPN on the hostile internet," in *Proceedings of the Free and Open Source Software Developers European Meeting-FOSDEM*, vol. 109, 2010. [Online]. Available: http://tinc-vpn.org/presentations/fosdem-2010/tinc_fosdem2010_slides.pdf

[12] FEIN Aachen e.V. Villasframework for distributed co-simulation. [Online]. Available: https://www.fein-aachen.org/projects/villas

[13] V. Jacobson, R. Frederick, S. Casner, and H. Schulzrinne. RTP: A transport protocol for real-time applications. [Online]. Available: https://tools.ietf.org/html/rfc3550

[14] C. Perkins, "Building adaptive applications: on the need for congestion control," in *Image and Video Communications and Processing 2005*, vol. 5685. International Society for Optics and Photonics, 2005, pp. 595–602. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/5685/0000/Building-adaptive-applications-on-the-need-for-congestion-control/10.1117/12.592797.short

[15] R. Liu, M. Mohanpurkar, M. Panwar, R. Hovsapian, A. Srivastava, and S. Suryanarayanan, "Geographically distributed real-time digital simulations using linear prediction," *International Journal of Electrical Power & Energy Systems*, vol. 84, pp. 308 – 317, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0142061516308079

[16] E. Guillo-Sansano, A. J. Roscoe, C. E. Jones, and G. M. Burt, "A new control method for the power interface in power hardware-in-the-loop simulation to compensate for the time delay," in *2014 49th International Universities Power Engineering Conference (UPEC)*, Sep. 2014, pp. 1–5.

[17] FEIN Aachen e.V. Co-simulation interfaces. [Online]. Available: https://www.fein-aachen.org/projects/cosif