A Co-Simulation Based Parallel and Multistep Approach for Accelerating EMT Simulations

M. Cai, J. Mahseredjian, H. Gras, A. El-Akoum, X. Fu

***Abstract*--This paper is related to research on parallelization and multistep techniques for the simulation of electromagnetic transients (EMTs). It presents latest improvements of a previously proposed approach based on the Functional Mock-up Interface standard adopting the concept of co-simulation for power grids with complex control systems.**

The control systems, decoupled in memory from the power network and encapsulated in Functional Mock-up Units, are represented by slave subsystems. In the improved approach, the subsystems are simulated either in parallel (asynchronous mode) or in sequence (synchronous mode), in which both modes accommodate the use of different time-steps in different decoupled subsystems. Extrapolation can also be effectuated in a sequential multistep simulation environment for enhanced simulation accuracy. Substantial computation time gains in both modes have been achieved in large-scale network protection studies with maintained accuracy.

***Keywords*: Electromagnetic transients, FMI, parallel simulation, multistep simulation, co-simulation.**

I. Introduction

THE Electromagnetic Transient (EMT) simulation tools have become indispensable for power system engineers [1]-[3]. The EMT-type simulation methods are circuit based and usually require significant computing time to solve large differential and algebraic equation (DAE) systems when compared to phasor domain methods. The numerical integration time-step, on the other hand, can also pose a constraint for simulation efficiency, even for smaller networks. Furthermore, more and more challenging cases, especially those with HVDC systems and wind generation [4], are created for the studies of modern-day power systems. All of these have prompted the research on computing time reduction for the simulation of electromagnetic transients and enhancing the performance of EMT-type solvers.

Many techniques have been proposed over the years to improve solution speed in EMT-type simulation methods, which include parallelization-based approaches (waveform relaxation [5], real-time techniques [6]-[8], and off-line [1], [9], [10] simulation tools) as well as multistep solution techniques (exploitation of circuit latency [11], and "data-smoothing" at line-bus interfaces [12]). Other techniques to enhance computational performance in EMT-type solvers include circuit reduction [13], frequency domain fitting [14] and approaches using dynamic equivalents [15].

This paper presents latest improvements of a previously proposed co-simulation-based off-line parallel and multistep approach [16] using the Functional Mock-up Interface (FMI) standard [17] for EMT simulations with complex control systems on conventional multi-core computers.

In the previously proposed approach [16], full compatibility between EMTP [1] and the FMI standard version 2.0 was established and the computation of control systems which are decoupled from the power network in memory is distributed among separate logical processors. Although two simulation modes, the *parallel asynchronous mode* and the *sequential synchronous mode*, were available, only the latter could accommodate the use of different time-steps in different decoupled subsystems. Continuing with the master-slave co-simulation scheme adopted in the previous approach, in the improved approach presented in this paper, the capacity of the *parallel asynchronous mode* has been extended into multistep simulation, further enhancing simulation efficiency and flexibility. Additionally, linear extrapolation has also been implemented in a sequential multistep simulation environment to achieve higher simulation accuracy. Apart from the various advantages over aforementioned techniques [5]-[10] explained in [16], the improved approach presented in this paper demonstrates even greater simulation efficiency in large-scale protection system studies thanks to its ability of accommodating multistep simulation in both modes and a higher level of accuracy with the incorporation of linear extrapolation.

The presented approach is applicable to any EMT-type simulation tool [3].

II. Master -Slave Co-simulation Using FMI

The implementation of the FMI standard in an EMT-type simulation tool was detailed in [16]. Therefore, this section only presents new improvements for the approach in [16] using the same EMTP software [1] for demonstrations.

A. Synchronization Between Master and Slaves in Different Simulation Modes

As in [16], synchronization between master and slave subsystems is achieved through the implementation of low-level synchronization primitive semaphores [18] in the DLL files FMI2_Link_Master and FMI2_Link_Slave with the help of pre-built functions *Wait* and *Release* that can be used by any process or thread.

The algorithms for the synchronous mode with the same

M. Cai, J. Mahseredjian, and H. Gras are with Polytechnique Montréal, QC, Canada, H3T1J4 (email of corresponding author: ming.cai@polymtl.ca).
A. El-Akoum is with Électricité de France (EDF) Lab Paris-Saclay, 7 Boulevard Gaspard Monge, 91120 Palaiseau, France.
X. Fu is with Tianjin University, 92 Weijin Road,Nankai District,Tianjin, China.

Paper submitted to the International Conference on Power Systems Transients (IPST2019) in Perpignan, France, June 18-21, 2019.

and different time-steps have been reorganized and combined in the improved approach as opposed to [16]. Fig. 1 illustrates the synchronization scheme between master and slave in the synchronous mode in the improved approach. It is noted that the FMI function fmi2DoStep is in charge of synchronization with the slave(s), and SemMaster and SemSlave represent semaphores released by the slave and master respectively. In the synchronous mode, after writing onto the co-simulation bus, the master first compares the current time-points of the master and the slave. If the master lags behind the slave, which occurs when $\Delta t_{master} < \Delta t_{slave}$, fmi2DoStep terminates immediately and the master continues its subsequent calculations until its current simulation time-point catches up with that of the slave. On the contrary, if the slave lags behind the master with $\Delta t_{master} \geq \Delta t_{slave}$, the master keeps releasing SemSlave to the slave whilst staying in the loop in order that the slave continues advancing in time and performing its calculations until it no longer lags behind the master. The master thereby retrieves the outputs from the slave and continues its own execution. Hence, the calculations of master and slave(s) are executed in sequence.

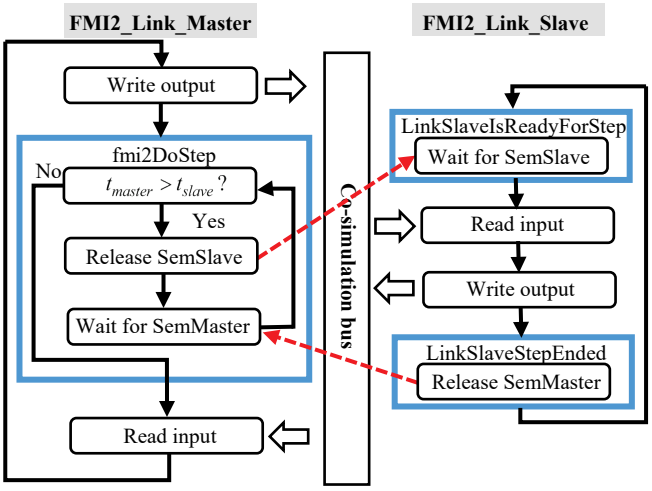


Fig. 1. Synchronization scheme between master and slave in the synchronous mode in the improved approach.

The asynchronous mode in the improved approach is divided into 3 cases, indicated by the three blue boxes, with only one case executed in practice based on the comparison of master and slave time-steps, as is shown in Fig. 2.

If $\Delta t_{master} = \Delta t_{slave}$, this is the case implemented in the previously proposed approach, as explained in [16].

If $\Delta t_{master} < \Delta t_{slave}$, after waiting for the slave to release SemMaster (green arrow), the master releases SemMaster (curved green arrow) to itself if it lags behind the slave in order to catch up with the latter. Once the master catches up with the slave, it immediately releases SemSlave (green arrow) to the slave such that both can execute their subsequent computations simultaneously.

If $\Delta t_{master} > \Delta t_{slave}$, as is illustrated in the third case, the master first waits for the slave to release SemMaster (blue arrow) after the slave completes its previous calculations. It then compares its current time-point with that of the slave. If the slave lags behind the master, the master keeps releasing SemSlave (blue arrow) to the slave such that the latter can catch up. Once the slave catches up, the master releases SemSlave (blue arrow) one more time and both continue their subsequent calculations in parallel.

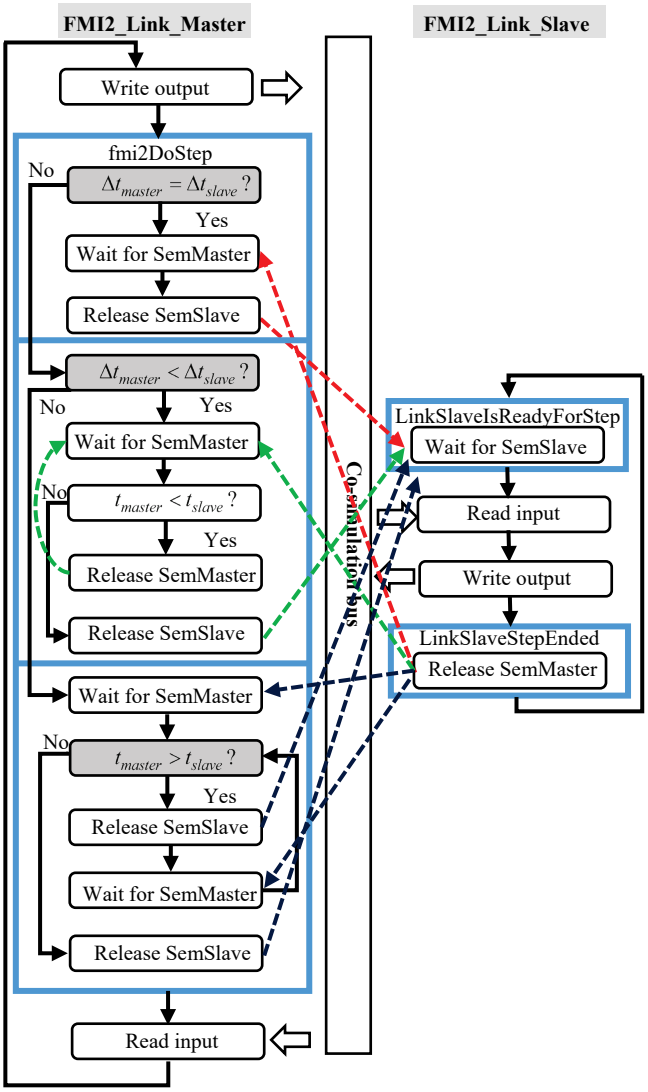


Fig. 2. Synchronization scheme between master and slave in the asynchronous mode in the improved approach.

B. Extrapolation of Floating-Point Type Signals in Sequential Multistep Simulation

Floating-point type signals from slave subsystems employing a larger time-step can also be extrapolated using linear extrapolation in the master for enhanced simulation accuracy in a sequential multistep simulation environment. This procedure is performed in the pre-built function *getFMUOutputs* whose main functionality is to read data exported from the slave, indicated in the FMI2_Link_Master as "Read input", as is shown in Fig. 1.

As is illustrated in Fig. 3, vector $\mathbf{y}_{slave}$ denotes floating-point type output from the slave whose size is n (number of floating-point type signals) at $t = T$, and vector $\mathbf{y}_{extrapol}$ of

size $2n$ stores floating-point type outputs from the slave at the previous two slave time-points ($t = T - \Delta t_{slave}$ and $t = T - 2\Delta t_{slave}$) used for extrapolation,

$$\mathbf{y}_{extrapol} = [\overbrace{\mathbf{y}_{slave,T-\Delta t_{slave}}}^{n}, \overbrace{\mathbf{y}_{slave,T-2\Delta t_{slave}}}^{n}] \quad (1)$$

Before extrapolating the slave output $\mathbf{y}_{slave}$ at $t = T$ using data stored in $\mathbf{y}_{extrapol}$, the function *getFMUOutputs* first compares the contents of $\mathbf{y}_{slave}$ and the first n elements of $\mathbf{y}_{extrapol}$, which is the slave output at $t = T - \Delta t_{slave}$. If these two terms are identical, indicating the slave, having a larger time-step, remains idle while the master keeps advancing and reading the same output from the slave as the previous slave time-point, a linear extrapolation is then performed using the slave outputs at the previous two slave time-points stored in $\mathbf{y}_{extrapol}$. On the contrary, if the contents of $\mathbf{y}_{slave}$ and the first n elements of $\mathbf{y}_{extrapol}$ differ from one another, which means the slave has advanced one step in time, no extrapolation is hence needed whereas the contents of $\mathbf{y}_{extrapol}$ are updated.

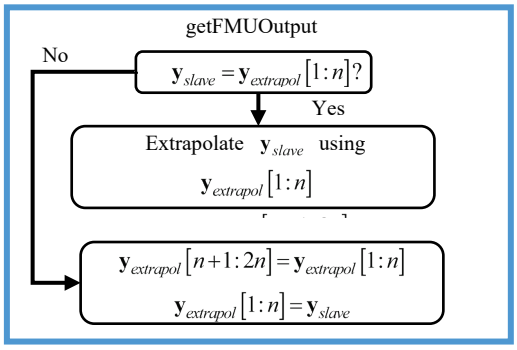


Fig. 3. Extrapolation of floating-point type signals in multistep simulation in the improved approach.

III. Test Cases

The new improvements proposed in this paper have been validated in terms of accuracy and simulation efficiency on practical system benchmarks. Accuracy is verified by comparing line relay tripping instants after a perturbation (fault) and voltage waveforms in sequential multistep simulation incorporating signal extrapolation. The reference waveforms in both cases are those found from the original single-core benchmarks without co-simulation. The computation time gains are based on the comparison of the solution times of the benchmarks simulated with and without co-simulation on a PC with 16 cores (32 logical processors). In all cases, a multiphase unbalanced load-flow solution is performed for initializing the time-domain solution where a fault condition is simulated.

A. Accuracy Validation

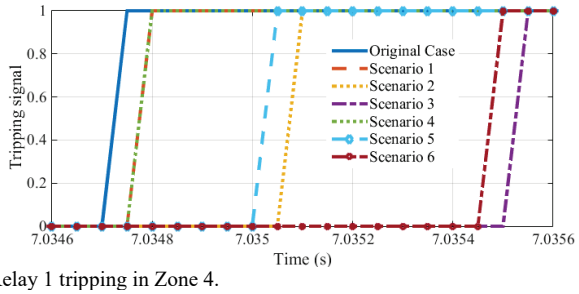
1) Test Case 1: Relay Tripping

The tests performed in this section use 6 co-simulation scenarios implementing the improved approach (with parallel multistep asynchronous mode) in which different co-simulation modes as well as different numerical integration time-steps were utilized in the control systems (line relays). This is shown in Table 1, where the Original Case is the original single-core benchmark without co-simulation.

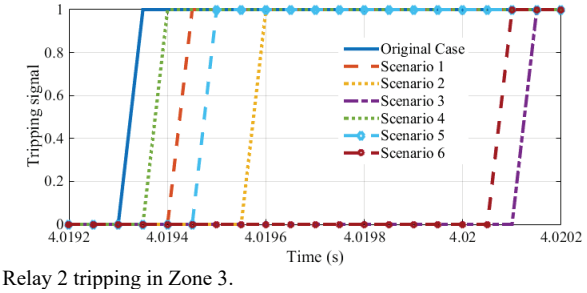
TABLE 1
CO-SIMULATION MODE SCENARIOS, RELAY TRIPPING

	Co-simulation mode	Main network Δt (μs)	Relay Δt (μs)
Original Case	N/A	50	50
Scenario 1	Asynchronous	50	50
Scenario 2	Asynchronous	50	200
Scenario 3	Asynchronous	50	400
Scenario 4	Synchronous	50	50
Scenario 5	Synchronous	50	200
Scenario 6	Synchronous	50	400

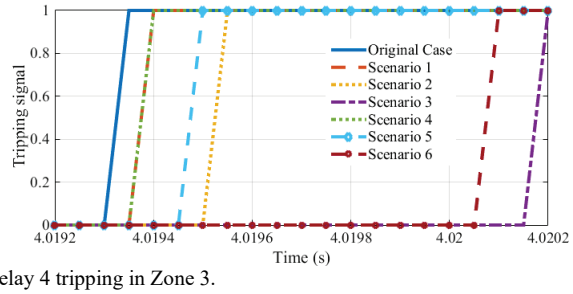
These tests validate the accuracy of co-simulation in relay applications by examining line relay tripping instants at adjacent lines after a phase-to-ground fault occurs at a line whose own relays are defective. The Network-2 benchmark in [16] is used in this test case. It is the 459-bus T1 test case in [20] including 7 voltage levels in transmission, sub-transmission, distribution and generation (see Figure 4-15 in [20]) with a total number of 3407 network nodes and the size of the main network equations being 12120×12120 (125120 non-zeros). Two line relays are added to every CP line in the benchmark. Fault location and relay settings can be found in [16]. The simulation interval is 10 s for all scenarios as in the Original Case, and the fault occurs at $t = 2s$.



(a) Relay 1 tripping in Zone 4.



(b) Relay 2 tripping in Zone 3.



(c) Relay 4 tripping in Zone 3.

Fig. 4. Tripping instants of Relay 1, 2 and 4.

The tripping instants of Relays 1, 2 and 4 are shown in Fig. 4. Detailed error analysis can be found in [16]. The largest relay tripping instant delay observed in this test case is 850 µs (Relay 4 in Scenario 3) compared to the Original Case, which is merely 1/19.6 cycle approximately. Catering to our need for simulation speedup and considering that the total clearing time (relay plus circuit breaker) typically ranges from 2 to 8 cycles on a 60 Hz base [19], this error is acceptably small, demonstrating that the accommodation of different time-steps in the parallel asynchronous mode does not affect accuracy.

2) Test Case 2 : Signal Extrapolation

This test case validates the implementation of extrapolation on floating-point type signals in a sequential multistep simulation environment in the improved approach. It uses two scenarios with different time-steps used in the three-phase voltage source (modeled using control devices), as is shown in Table 2 in which the Original Case is the original single-core benchmark without co-simulation.

This test was performed on the benchmark shown in Fig. 5. It is a simple 5-bus 12.5 kV system with one three-phase voltage source modeled using control devices. Other network components and device parameters can be seen in Fig. 5. The simulation interval is 1 s for both co-simulation scenarios as in the Original Case. At $t = 0.5s$, breakers BR1 and BR2 open, detaching loads FdrLoad and Load4801 as well as transformer TRANSFO from the system. In both scenarios, the three-phase voltage source is decoupled from the electrical network to form the slave subsystem.

TABLE 2
CO-SIMULATION MODE SCENARIOS, SIGNAL EXTRAPOLATION

	Co-simulation mode	Main network Δt (µs)	Source Δt (µs)
Original Case	N/A	10	10
Scenario 1	Synchronous	10	50
Scenario 2	Synchronous	10	100

Fig. 6 and Fig. 7 present the phase-b voltage at bus XFMR in the Original Case and both co-simulation scenarios without and with signal extrapolation respectively. For clear observation, only the interval between $t = 0.5s$ and $t = 0.55s$ after circuit breaker operation is shown in Fig. 6 (a) and Fig. 7 (a).

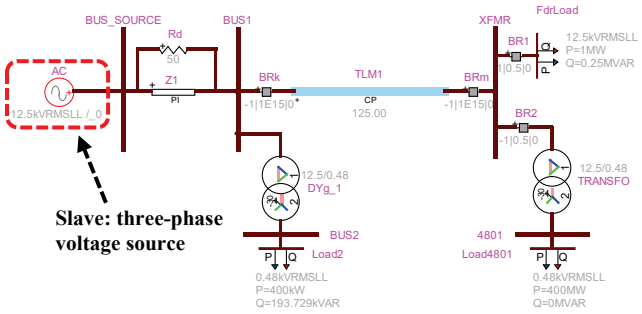
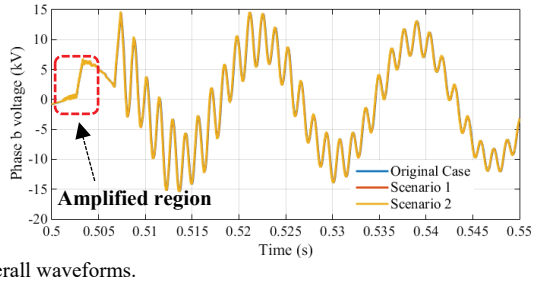
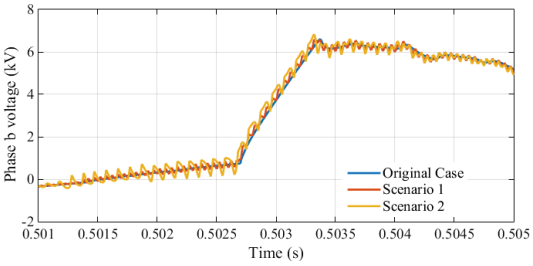


Fig. 5. Signal extrapolation test benchmark.

Without signal extrapolation, although the voltage waveform transients in the two co-simulation scenarios generally agree with that obtained in the Original Case, as can be seen in Fig. 6 (a), a closer observation in the amplified region clearly indicates that the waveforms in the two co-simulation scenarios exhibit a "sawtooth" characteristic. This is due to the fact that the master keeps reading the same output from the slave as the former advances whilst the slave (three-phase voltage source in this test case), having a larger time-step, remains idle (please refer to the Appendix in [16]).

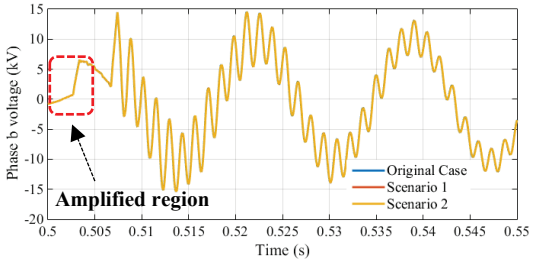


(a) Overall waveforms.

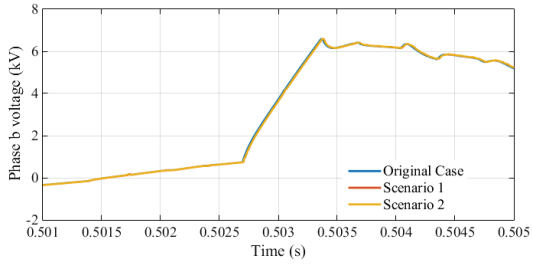


(b) Waveforms in the amplified region.

Fig. 6. Phase-b voltage at bus XFMR between $t = 0.5s$ and $t = 0.55s$ without signal extrapolation with closer observation of waveforms in the amplified region.



(a) Overall waveforms.



(b) Waveforms in the amplified region.

Fig. 7. Phase-b voltage at bus XFMR between $t = 0.5s$ and $t = 0.55s$ with signal extrapolation with closer observation of waveforms in the amplified region.

However, with the implementation of signal extrapolation on the slave outputs, this "sawtooth" characteristic can be largely smoothed out, allowing the waveforms from the two co-simulation scenarios to better agree with that obtained in the Original Case, as can be observed in Fig. 7. It is noted that the extra one time-step delay added during master slave data exchange for simulation stability and robustness still exists.

It can thus be concluded that better simulation accuracy can be achieved by the implementation of floating-point type signal extrapolation in a sequential multistep simulation environment.

B. Computation Time Gains

The objective of this section is to demonstrate the numerical performance advantages of the improved approach with parallel multistep asynchronous mode on practical power system networks of different levels of complexities, Network-1 (see [16], also Figure 2-1 and Figure 2-8 in [20]) and -2 (see also Figure 4-15 in [20]). The Network-1 benchmark is the IEEE-39 test case including wind generation proposed in [20] with a total of 1381 network nodes. The size of its main network equations is 4826×4826 with 35718 non-zeros. The hardware and software configurations of the PC used in the tests are given in Table 3.

TABLE 3
PC HARDWARE AND SOFTWARE CONFIGURATIONS

Processor	Intel (R) Xeon (R) E5-2667 v3
# of cores	16
# of logical processors	32
Operating system	Microsoft Windows 10 Pro

The tests include the Original Case without co-simulation and 5 simulation scenarios (see Table 4) implementing the parallel multistep asynchronous mode in the improved approach, where different co-simulation modes and numerical integration time-steps were employed in the control systems (offshore wind park controls and line relays). A simulation interval of 2 s is used in all test cases.

TABLE 4
CO-SIMULATION MODE SCENARIOS, COMPUTATION TIME GAINS

	Co-simulation mode	Main network Δt (μs)	WP control Δt (μs)	Relay Δt (μs)
Original Case	N/A	50	50	50
Scenario 1	Asynchronous	50	50	50
Scenario 2	Asynchronous	50	200	200
Scenario 3	Asynchronous	50	200	400
Scenario 4	Synchronous	50	200	200
Scenario 5	Synchronous	50	200	400

1) Test Case Network-1

Benchmark Network-1 (IEEE-39 test case including wind generation), shown in Figure 2-1 and Figure 2-8 in [20], was used in the first batch of tests. Benchmark description and device modelling details are also presented in [20]. Different numbers of line relays, presented in Fig. 9 in [16], were added into this benchmark (see Table 5). The slave subsystems in this test case thus consist of all 4 offshore aggregated wind park control systems (average model, see [4]) and line relays implemented using co-simulation in all scenarios.

The total solution times for Network-1 in different scenarios compared to the Original Case are presented in Table 5, and the solution time speedup in comparison to the Original Case with respect to the number of slaves (offshore wind park controls and line relays) is shown in Fig. 8.

2) Test Case Network-2

The second batch of tests were performed on Network-2 benchmark (T1 test case) shown in Figure 4-15 in [20] with different numbers of line relays used in each test (see Table 6). Offshore wind parks are not present in this test case. Therefore, the slave subsystems in this test case are only comprised of line relays in all scenarios.

The total solution times for Network-2 in different scenarios compared to the Original Case are presented in Table 6, and the solution time speedup in comparison to the Original Case with respect to the number of slaves (line relays) is shown in Fig. 9.

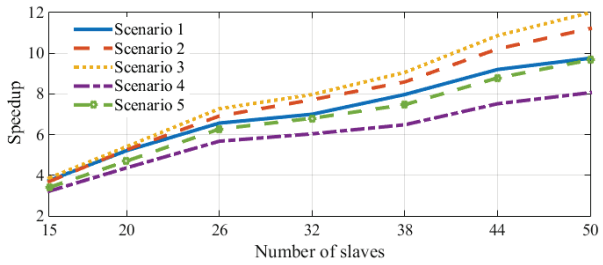


Fig. 8. Total solution time speedup with respect to the number of slaves in Network-1.

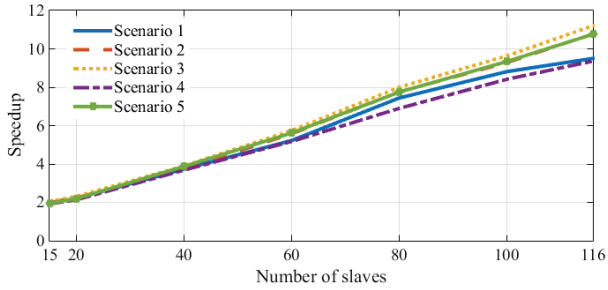


Fig. 9. Total solution time speedup with respect to the number of slaves in Network-2.

Overall, it can be observed in Fig. 8 and Fig. 9 that substantial solution time speedup is obtained in power system transient simulations using the improved approach proposed in this paper. In both test cases, it is observed that the asynchronous mode with different time-steps renders the highest speedup, a speedup that was not observed previously in [16], confirming that extending the capacity of the parallel asynchronous mode into multistep simulation further enhances simulation efficiency.

Apart from the conclusions already drawn regarding the speedup achieved in different modes (see [16]), it is also worth mentioning that the results presented in Fig. 8 and Fig. 9 do not clearly show speedup saturation although the number of decoupled subsystems in some cases far exceeds the number of cores and logical processors of the PC. This is because: firstly, all control systems are decoupled from the main network in memory; secondly, the equations of each decoupled control system are formulated and solved individually on separate processors, which is far more computationally efficient than formulating the equations of all control systems into one large matrix as in the case without co-simulation. The high scalability in the improved approach indicates even higher speedup could be achieved in protection studies of large-scale networks using even more and different types of protection relays and control systems.

TABLE 5
COMPARISON OF SOLUTION TIMES (S) ON THE PC WITH 16 CORES FOR DIFFERENT SCENARIOS, NETWORK-1

Number of WP controls	4						
Number of relays	11	16	22	28	34	40	46
	Total solution time (s)						
Original Case	824	1292	1963	2415	2989	3901	4644
Scenario 1	220	248	299	345	375	424	476
Scenario 2	223	246	284	313	348	382	414
Scenario 3	213	239	270	303	330	359	387
Scenario 4	256	296	346	400	461	519	576
Scenario 5	242	275	313	355	400	444	480

TABLE 6
COMPARISON OF SOLUTION TIMES (S) ON THE PC WITH 16 CORES FOR DIFFERENT SCENARIOS, NETWORK-2

Number of relays	15	20	40	60	80	100	116
	Total solution time (s)						
Original case	2419	2840	5494	9288	14426	19249	24351
Scenario 1	1237	1287	1471	1778	1939	2184	2563
Scenario 2	1214	1271	1461	1665	1860	2073	2259
Scenario 3	1191	1228	1409	1617	1804	1998	2174
Scenario 4	1259	1328	1492	1795	2091	2288	2601
Scenario 5	1246	1294	1418	1654	1862	2061	2263

IV. CONCLUSION

This paper presented latest developments of a previously proposed co-simulation-based parallel and multiple time-step approach using the Functional Mock-up Interface standard for the simulation of power system transients.

The algorithm of the sequential synchronous mode has been reorganized and the capacity of the parallel asynchronous mode has been extended into accommodating the use of different time-steps in different decoupled subsystems, further enhancing simulation efficiency and flexibility. Moreover, linear extrapolation has been implemented in the synchronous mode with different time-steps on floating-point type control signals, which greatly improves simulation accuracy. The improved approach has been tested in terms of accuracy and solution speedup using realistic power system simulation benchmarks, validating the new developments for the previously proposed approach in these two aspects.

V. REFERENCES

[1] J. Mahseredjian, S. Dennetière, L. Dubé, B. Khodabakhchian, and L. Gérin-Lajoie, "On a new approach for the simulation of transients in power systems," *Electric power systems research*, vol. 77, no. 11, pp: 1514-1520, 2007.

[2] A. Ametani, "Numerical Analysis of Power System Transients and Dynamics", IET (The Institution of Engineering and Technology), 2015.

[3] J. Mahseredjian, V. Dinavahi, and J.A. Martinez "Simulation Tools for Electromagnetic Transients in Power Systems: Overview and Challenges," *IEEE Transactions on Power Delivery*, vol. 24, issue 3, pp. 1657-1669, July 2009.

[4] U. Karaagac, J. Mahseredjian, L. Cai, and H. Saad, "Offshore Wind Farm Modeling Accuracy and Efficiency in MMC-Based Multi-Terminal HVDC Connection," *IEEE Trans. on Power Delivery*, vol. 32, no. 2, pp. 617-627, April 2017.

[5] J. M. Bahi, K. Rhofir, and J.-C. Miellou, "Parallel solution of linear DAEs by multisplitting waveform relaxation methods," *Elsevier, Linear Algebra and its Applications*, vol. 332-334, pp. 181-196, Aug. 2001.

[6] D. Paré, G. Turmel, J.-C. Soumagne, V. Q. Do, S. Casoria, M. Bissonnette, B. Marcoux, and D. McNabb, "Validation tests of the Hypersim digital real time simulator with a large AC-DC network," in *International conference on power system transients (IPST)*, New Orleans, USA, 2003.

[7] S. Abourida, C. Dufour, J. Belanger, G. Murere, N. Lechevin, and B. Yu, "Real-time PC-based simulator of electric systems and drives," in *Proc. 17th IEEE APEC, Applied Power Electronics Conf. and Expo*., pp. 433-438, Dallas, USA, Mar. 10-14, 2002.

[8] R. Kuffel, J. Giesbrecht, T. Maguire, R. P.Wierckx, and P. G. McLaren, "RTDS-A fully digital power system simulator operating in real-time," in *Proc. WESCANEX*, vol.2, pp. 300-305, Winnipeg, Manitoba, Canada, 1995.

[9] R. Singh, A. M. Gole, C. Muller, P. Graham, R. Jayasinghe, B. Jayasekera, and D. Muthumuni, "Using Local Grid and Multi-core Computing in Electromagnetic Transients Simulation," in *International conference on power system transients (IPST)*, Vancouver, Canada, 2013.

[10] S. Montplaisir-Goncalves, J. Mahseredjian, O. Saad, X. Legrand, and A. El-Akoum, "A Semaphore-based Parallelization of Networks for Electromagnetic Transients," in *International conference on power system transients (IPST)*, Vancouver, Canada, 2013.

[11] A. Benigni, A. Monti, and R. A. Dougal, "Latency-based approach to the simulation of large power electronics systems," *IEEE Transactions on Power Electronics*, vol. 29, no. 6, pp. 3201–3213, June 2014.

[12] A. Semlyen, and F. De Leon, "Computation of electromagnetic transients using dual or multiple time steps," *IEEE Transactions on Power Systems*, vol. 8, no. 3, pp. 1274-1281, Aug. 1993.

[13] U. N. Gnanarathna, A. M. Gole and R. P. Jayasinghe, " Efficient Modeling of Modular Multilevel HVDC Converters (MMC) on Electromagnetic Transient Simulation Programs," *IEEE Trans. on Power Delivery*, vol. 26, no. 1, pp. 316-324, Jan. 2011.

[14] B. Gustavsen, "Passivity enforcement of rational models via modal perturbation," *IEEE Transactions on Power Delivery*, vol. 23, pp. 768-775, Apr 2008.

[15] U. D. Annakkage, N. K. C. Nair, Y. Liang, A. M. Gole, V. Dinavahi, B. Gustavsen, et al., "Dynamic System Equivalents: A Survey of Available Techniques," *IEEE Transactions on Power Delivery*, vol. 27, pp. 411-420, 2012.

[16] M. Cai, H. Gras, J. Mahseredjian, E. Rutovic, and A. El-Akoum, "Functional Mock-up Interface-Based Approach for Parallel and Multistep Simulation of Electromagnetic Transients," *IEEE Trans. on Power Delivery* (Early Access).

[17] Modelica Association Project "FMI", "FMI for Model Exchange and Co-Simulation - Version 2.0," 2014.

[18] R. H. Arpaci-Dusseau and A. C. Arpaci-Dusseau, "Semaphores," in *Operating Systems: Three Easy Pieces*. Madison, WI, USA: Arpaci-Dusseau Books, 2014, pp.1-18.

[19] J. L. Blackburn and T. J. Domin, *Protective Relaying: Principles and Applications*, 3rd ed. Boca Raton, FL: CRC Press, 2007, pp. 47, 592.

[20] CIGRE Working Group C4.503, Power system test cases for EMT-type simulation studies. CIGRE Technical Report, Aug.