Real-Time Simulation of MMCs Using CPU and FPGA

Hani Saad, *Student Member, IEEE*, Tarek Ould-Bachir, *Member, IEEE*, Jean Mahseredjian, *Fellow, IEEE*, Christian Dufour, *Member, IEEE*, Sébastien Dennetière, *Member, IEEE*, and Samuel Nguefeu, *Member, IEEE*

***Abstract*—Modular multilevel converter (MMC) structures are composed of several hundreds to thousands of half-bridge converters. Such large numbers of power switches and electrical nodes introduce important numerical challenges for the computation of electromagnetic transients. The problem becomes particularly more complex for the real-time simulations. This paper presents a feasibility study on the real-time simulation of the MMC models. CPU-based and field-programmable gate array-based implementations are proposed and evaluated for the MMCs having up to 401 levels. The study also provides guidelines for the real-time simulation platform requirements to simulate these MMC models.**

***Index Terms*—Electromagnetic transients program (EMTP), field-programmable gate array (FPGA), high-voltage direct current (HVDC), modular multilevel converter (MMC), real time, voltage source converter (VSC).**

I. Introduction

THE high-voltage direct current (HVDC) systems in electric power grids are becoming increasingly popular [1]. The use of voltage source converters (VSCs) based on insulated gate bipolar transistors (IGBTs) is becoming more attractive, mainly due to their higher performances and cost [2]. The recent modular multilevel converter (MMC) topology based on the half-bridge modules connected in series [3] offers significant advantages over previous VSC technologies [4], [5].

Real-time simulation of power converters is a valuable tool for development and testing of control systems [6]. Real-time modeling of power electronics can be implemented mainly through CPUs and field-programmable gate arrays (FPGAs). CPU-based execution times for each time point remain in the range of tens of microseconds and constitute a limiting factor for the simulation accuracy. FPGA technology offers a better alternative and allows reducing computation times to the range of hundreds of nanoseconds. However, important concerns remain for the broad adoption of this technology due to significant efforts needed to implement complex solvers at a fixed point. The fixed-point format is inherent to FPGA design and was reported to limit the achievable calculation time-step because of its narrow range coverage and quantization issues [7]. The floating-point (FP) format, on the other hand, has the potential to popularize FPGAs, but the circuitry of FP operators and related long latency limit that potential by making small calculation time-steps almost out of reach for more complex applications. However, the use of custom FP operators and nonstandard FP formats on FPGA is a promising path [8].

Depending on application and power capability requirements, the MMC levels can vary from tens to hundreds of submodules (SMs) per arm. For the HVDC and FACTS systems, an MMC may include thousands of power switches. The Trans Bay Cable project [9], for example, includes more than 200 SMs per arm and the INELFE [10] installation will include more than 400 SMs per arm. The excessive number of MMC power switches creates significant computational difficulties when simulating electromagnetic transients [11]. The numerous nonlinear devices in the converter require an iterative solution which significantly increases the computational burden. In real-time simulations, modeling a highly detailed switching device is currently out of reach and simplifications are required to achieve hardware-in-the-loop (HIL) simulations. By simplifying each IGBT/diode device to the level of a switchable $R_{\rm ON}$/$R_{\rm OFF}$ resistance, the Norton equivalent of each MMC arm can be obtained [12]. In [13]–[15], the latter approach is implemented on the CPU for real-time simulations. However, there is no report in the literature on the successful real-time simulation of MMCs with hundreds of SMs per arm.

This paper presents the feasibility study of real-time simulation of MMC models having up to 400 SMs/arm. The models are implemented using both CPU and FPGA technologies in order to evaluate limitations and advantages of each technology for real-time simulation. This paper also provides guidelines on real-time platform architecture requirements according to the number of MMC levels.

This paper is structured as follows. Section II is used for a brief presentation of the MMC. Section III presents a relation between controller sampling time and the number of MMC levels. The MMC model used in this paper is briefly recalled in Section IV. Section V presents model implementation on the CPU and FPGA.

II. MMC Topology and Control System

Fig. 1(a) shows the three-phase configuration of the MMC topology. The MMC is comprised of N SMs per arm which yields a line-to-neutral voltage waveform of $(N+1)$ levels [16].

Manuscript received July 15, 2013; accepted September 10, 2013. Date of publication September 19, 2013; date of current version August 26, 2014. Recommended for publication by Associate Editor J. R. Espinoza.

H. Saad, T. Ould-Bachir, and J. Mahseredjian are with the Department of Electrical Engineering, École Polytechnique de Montréal, Montreal, QC H3T 1J4, Canada (e-mail: hani.saad@polymtl.ca; tarek.ould-bachir@polymtl.ca; jeanm@polymtl.ca).

C. Dufour is with the Opal-RT Technologies, Montréal, QC H3K 1G6, Canada (e-mail: christian.dufour@opal-rt.com).

S. Dennetière and S. Nguefeu are with the Réseau de Transport d'Electricité, Paris 92932, France (e-mail: sebastien.dennetiere@rte-france.com; samuel.nguefeu@rte-france.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TPEL.2013.2282600

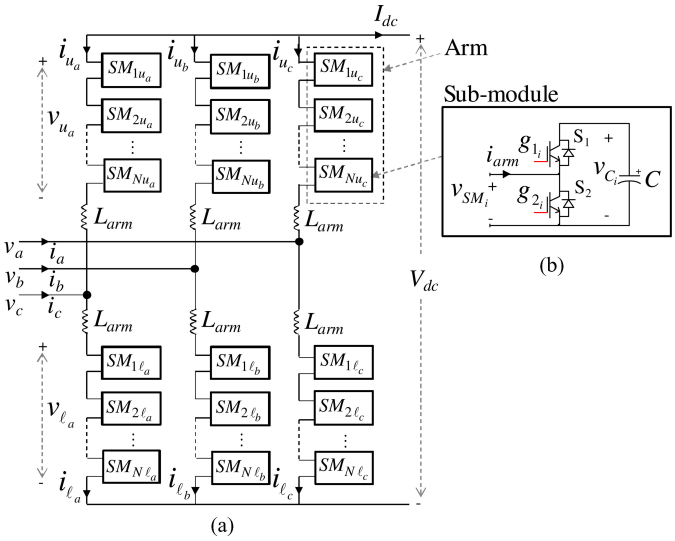

Fig. 1. (a) Typical MMC topology for a three-phase converter. (b) Half-bridge circuit for the ith SM.

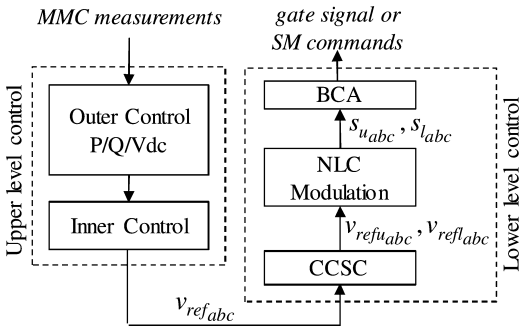

Fig. 2. Control hierarchy for the VSC-MMC station.

The inductor $L_{\rm arm}$ is added on each arm to limit arm current harmonics and fault currents. Each SM is a half-bridge converter as depicted in Fig. 1(b) and includes mainly a capacitor C and two IGBTs with antiparallel diodes (S_1 and S_2).

Since the IGBT device is controllable through gate signals g_{1_i} and g_{2_i}, the SM can have three different states. In the ON state, g_{1_i} is ON, g_{2_i} is OFF, and the SM voltage $v_{\rm SM_i}$ is equal to the capacitor voltage v_{C_i}. In the OFF state, g_{1_i} is OFF, g_{2_i} is ON, and $v_{\rm SM_i} = 0$. In the blocked state: g_{1_i} is OFF, g_{2_i} is OFF, and $v_{\rm SM_i}$ depends on the arm current $(i_{\rm arm})$ direction. The capacitor may charge through S_1 and cannot discharge.

The MMC topology is of VSC type [17] and uses an upper level control including an inner current controller and an outer controller. The MMC topology requires additional controllers in order to stabilize internal variables: SM capacitor voltages and second harmonic circulating currents of each phase [18]. This constitutes the lower level controller which includes the circulating current suppressing control, the nearest level control modulation (NLC), and the balancing control algorithm (BCA) of capacitor voltages. A top level view of the control structure is presented in Fig. 2. More details on the control system can be found in [10]. The control system output can be either the gate signal or the SMs command which is the binary addition of the SM gate signals $(g_{1_i}$ and $g_{2_i})$.

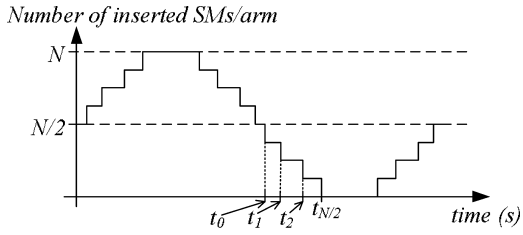

Fig. 3. Typical NLC output waveform for a MMC-9-level ($N = 8$).

III. Relation Between Sampling Time and N

The main objective of this study is to evaluate the feasibility and performance of the MMC model in the real-time simulation. The sampling time of the control system plays an important role IN the dynamic performance of the system. This sampling time is mainly determined by the used modulation technique. The phase-disposition pulse width modulation (PWM), phase-shifted carrier PWM [19], selective harmonic elimination PWM, and space-vector PWM are used for the lower MMC level in order to increase harmonic frequency. As N increases, PWM types become less efficient. Therefore, more efficient staircase-type methods, such as the NLC technique, have been proposed in [20]. The NLC modulation developed in this paper uses the round function in order to transform the reference voltage to a staircase waveform with the number of steps equal to MMC levels. In order to guarantee the passage through each level (as stated in [1]), the sampling time must respect certain criteria. A typical waveform produced by the NLC modulation for MMC-9-level $(N = 8)$ is shown in Fig. 3. The equation of this staircase waveform is given by

$$s_n = \frac{N}{2}\sin\left(2\pi f\ t_n\right) + \frac{N}{2} \tag{1}$$

where $t_n = t_0, t_1, \ldots, t_{N/2}, s_n$ is the number of SMs to be inserted in each arm, and f is the network frequency.

The smallest time interval between two different levels is found where the slope of the desired sine wave is the highest, that is between t_0 and t_1. Therefore, the controller sampling time-step Δt should respect the following inequality in order to guarantee the replication of each MMC level:

$$\Delta t \leq \frac{1}{2\pi f}\arcsin\left(\frac{2}{N}\right). \tag{2}$$

It is noticed that (2) does not account for variations produced by the control system. The amplitude related to *N/2* and f may vary due to the reference voltage variations. These parameters depend essentially on the MMC application. For the HVDC and FACTS applications, these parameters can be bounded, since frequency and amplitude of an ac grid cannot exceed practical limits. If we set a 1.2 pu higher limit for frequency and a 1.4 pu limit for amplitude, (2) yields

$$\Delta t \leq \frac{1}{1.2\,(2\pi f)}\arcsin\left(\frac{2}{1.4N}\right). \tag{3}$$

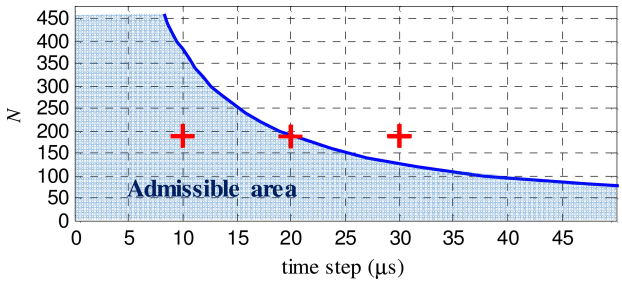

Fig. 4. N as a function of sampling time-step limit.

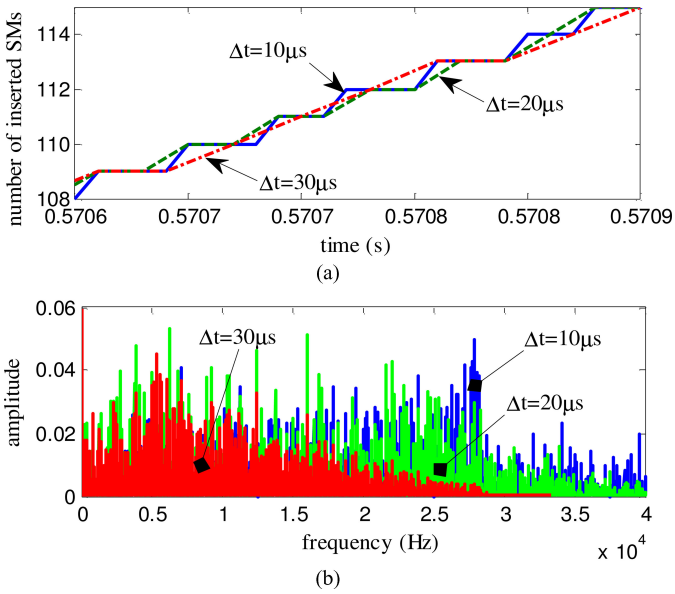

Fig. 5. Impact of Δt on the modulated waveform. (a) Accuracy of modulated waveform for different Δt cases. (b) FFT of modulated waveform for different Δt cases.

Equation (3) is shown in Fig. 4. The colored area under the curve represents the admissible zone in which Δt will guarantee the replication of each MMC level. As expected, when N increases, the Δt limit decreases.

In order to analyze the impact of Δt, an NLC modulation for a MMC-191-level ($N = 190$) is simulated with three different Δt values highlighted with a red cross in Fig. 4 ($\Delta t = 10$, 20 and 30 μs). Fig. 5 shows the modulated waveforms and resulting harmonics for different values of Δt. For sampling times that respect (3) ($\Delta t = 10$ and 20 μs), the passage through each MMC level is guaranteed and the harmonic contents are sufficiently close. With $\Delta t = 30$ μs, it is not possible to reproduce all levels and the harmonic content is significantly affected. The resulting level jumps will also impact the switching frequency of each SM and the BCA.

Equation (3) is valid for the NLC modulation technique and different formulas must be derived for other modulation techniques.

IV. MMC Model

The MMC model used in this paper has been presented in [15] and only a brief description is recalled later. In this model, the

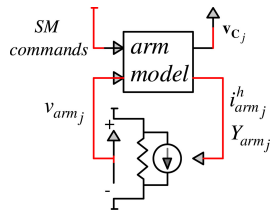

Fig. 6. MMC jth arm model.

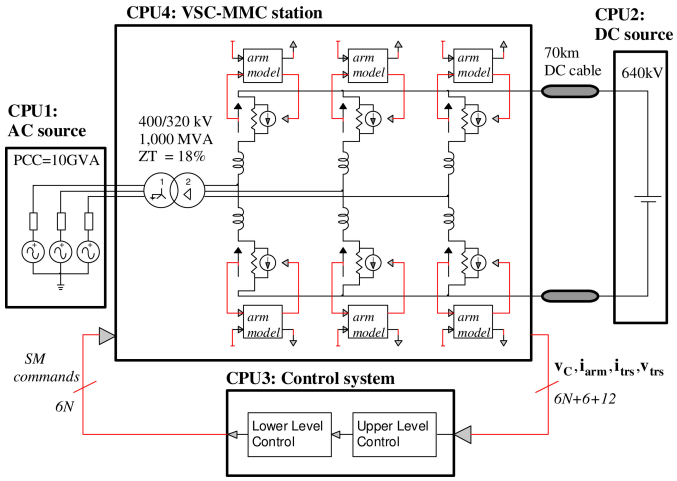

Fig. 7. MMC circuit and one-CPU MMC model structure.

SM power switches are replaced by ON/OFF resistors: R_{ON} (small value in mΩ) and R_{OFF} (large value in MΩ). This approach allows performing an arm circuit reduction for eliminating internal electrical nodes and allowing the creation of a Norton equivalent for each MMC arm [12]. The model still considers each SM separately and maintains a record for each individual capacitor voltage. The schematic structure of the model is depicted in Fig. 6. Following the network solution, the algorithm receives the voltages of each arm j through v_{arm_j} and sends Norton equivalent values Y_{arm_j} and $i^h_{\mathrm{arm}_j}$. Because the model is implemented using the SimPowerSystems tool for Simulink, the state-space nodal approach [21] is used to interface state-space network equation with the nodal equation of each arm. The interface between the arm model and the control system is similar to a detailed MMC model; it receives the SM commands and sends back the SM capacitor voltages through the vector $\mathbf{v}_{C_j}$.

V. Real-Time Simulation

The studied case is presented in Fig. 7. The control strategy considers an active/reactive power flow control. The ac grids are represented as equivalent sources with a short-circuit level of 10 000 MVA. The transmission capacity of the system is 1000 MW. The dc side is represented with a dc cable modeled using the distributed parameter cable (DPC) model (constant parameters) and a dc voltage source of 640 kV. The simulations were performed on the OP5600, the real-time simulator from Opal-RT. The OP5600 simulator has three quad-core processors (for a total of 12 Intel Xeon CPUs) that communicate through a

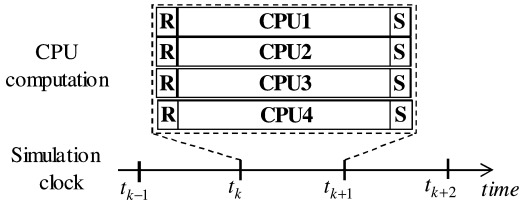

Fig. 8. Real-time computations for one-CPU MMC model.

shared memory of 8 GB, and that are able to communicate with the FPGA board through a second generation PCIe link.

A. CPU-Based Model

This section presents a feasibility analysis using the CPU technology. To simulate an electrical network in real time, it is necessary to decouple the network on a multiprocessor system in order to parallelize and accelerate the computations.

1) One-CPU MMC: A logical separation is presented in Fig. 7: CPU1 for the equivalent network, CPU2 for the dc voltage source, CPU3 for the control system, and CPU4 for the MMC station.

Large amount of data is exchanged between CPU3 and CPU4. All SM commands ($6N$) are sent from CPU3 to CPU4. The capacitor voltages $\mathbf{v_C}$ of all SMs ($6N$), the arm currents $\mathbf{i_{arm}}$ (6), and transformer primary and secondary currents $\mathbf{i_{trs}}$, and voltages $\mathbf{v_{trs}}$ (12) are transmitted from CPU4 to CPU3. Stub-line blocks are used between CPU1 and CPU4 to decouple the processed equations. The DPC model decouples CPU4 and CPU2. Further details about these decoupling techniques can be found in [22].

The complete circuit is parallelized as shown in Fig. 8, and the R and S blocks represent receiving and sending data functions, respectively.

Real-time simulation accuracy verification for this MMC model is presented in [15]. The comparison against a detailed MMC model version, where each power switch is modeled as a nonlinear device (detailed IGBT and diode models), shows a close match with a relative error of less than 5%. Thus, in this paper, the one-CPU MMC model is taken as the reference model to verify all other configurations presented in the following sections.

2) Multi-CPU MMC: In order to improve computational performance, each arm model is solved on its own CPU (see Fig. 9). In this topology, CPU4 contains only the Norton equivalent circuits. Parallelization now requires the introduction of an artificial one time-step delay, which is not acceptable since the Norton equivalent circuit should be computed within the same time-step [21]. The artificial delay is avoided by computing the arm models and the Norton equivalent circuit in series. Nevertheless, since each arm code is computed independently, parallelization between arms is still possible. Thus, the MMC model uses the series/parallel computation scheme depicted in Fig. 10.

3) Real-Time CPU-Based Model Accuracy Verification: An evaluation of the multi-CPU model performance in the real-time simulation is performed here and compared against the

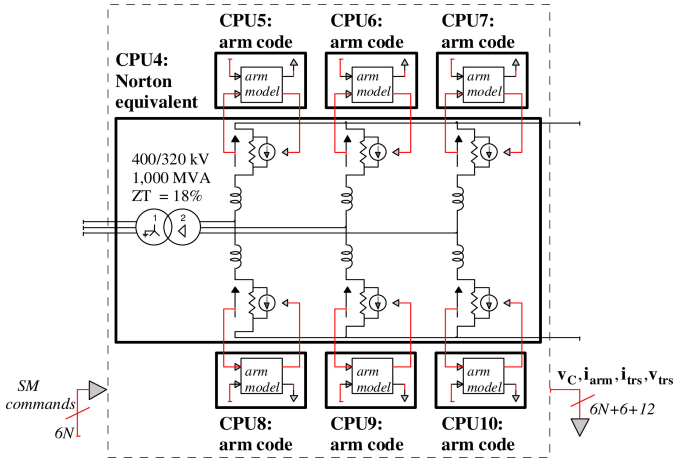

Fig. 9. Structure of the multi-CPU MMC model.

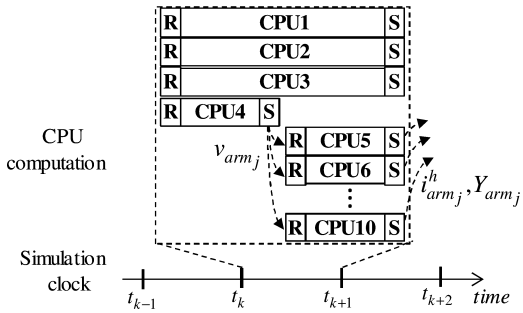

Fig. 10. Real-time computation for the multi-CPU MMC model.

one-CPU model (reference model). In all figures, a blue solid line is used for the reference model and a green dotted line for the multi-CPU MMC model. An MMC-101-level is considered with a time-step of 30 μs. A 150 ms three-phase-to-ground fault is applied on the ac side (between CPU1 and CPU4) at 10 s of simulation time. Figs. 11 and 12 compare the dynamic responses of several MMC variables. It is apparent that a close match is achieved between the two CPU configurations.

The differences observed between $v_{C_1 u_a}$ [see Fig. 12(c)] are related to the capacitor balancing control strategy of the BCA. This control selects the optimal SM number that has to be switched. These selections are not necessarily identical between two simulations. However, the sum of all capacitor voltages of each arm must be identical for both models, which is the case, as shown in Fig. 12(d).

The relative errors in compared variables vary between 0.1 and 2%. Similar results are obtained for other test cases.

4) Real-Time Performances of CPU-Based Models: The real-time performances of one-CPU and multi-CPU configurations are analyzed by varying N from 60 to 400. For both configurations, the execution times of CPU1 and CPU2 are not affected by N and remain very low (1 μs and 2.1 μs, respectively). On the other hand, the control system CPU3 is affected by the increase in N and this is mainly due to the presence of the BCA. The execution time ratio between CPU3 and MMC circuit CPUs (CPU5 to CPU10 of Fig. 9) is less than 0.5, which proves

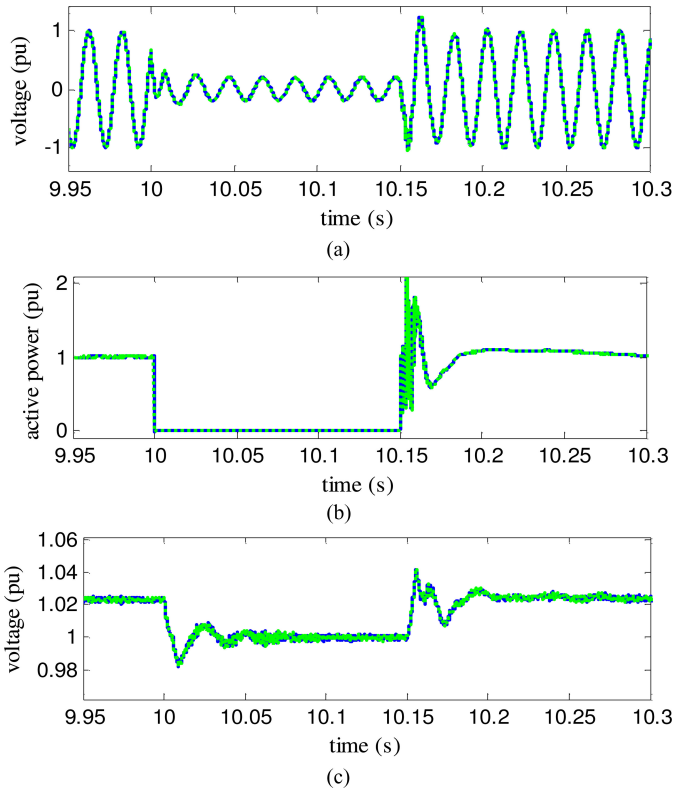

Fig. 11. MMC external variables, reference and multi-CPU MMC models. (a) Phase voltage: v_a. (b) Active power: P. (c) DC voltage: V_{dc}.

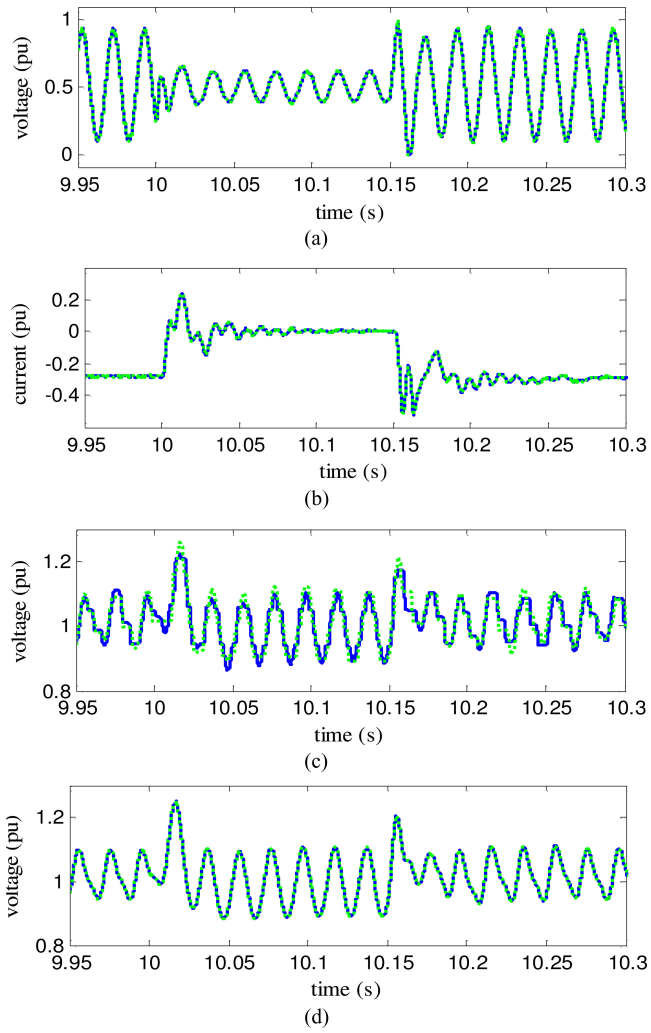

Fig. 12. MMC internal variables, reference, and multi-CPU MMC models. (a) Arm voltage: v_{u_a}. (b) Circulating current: $(i_{u_a} + i_{l_a})/2$. (c) Capacitor voltage: $v_{C_1 u_a}$. (d) Sum of all capacitor voltages: $\sum_{i=1}^{N} v_{C_i u_a}$.

that the MMC circuit CPUs constitute the main bottleneck for real-time simulation.

Fig. 13 presents N as a function of execution time for CPU4 in Fig. 7 (one-CPU setup). As depicted in [15], the execution time increases linearly with the number of SMs and crosses the NLC sampling time limit for the real-time simulation at $N = 120$. Fig. 13 also shows the execution time results for the multi-CPU MMC configuration. Parallelization between arm models reduces the execution time, but the gain is not six times, since the CPU4 in Fig. 9 is in series with CPUs 5 to 10 (see Fig. 10). For 100 SMs per arm, the mean execution time for the multi-CPU MMC setup is ≈12.3 μs, whereas the one-CPU MMC version uses 26.4 μs, which gives a ratio of 2.15. This ratio increases with the number of SMs. The maximum number of SMs/arm that can be simulated in the real-time with the multi-CPU MMC setup is about 230.

If the CPU-based models are to be used in an HIL configuration, where a real control system is interfaced with the numerical simulator, the latency produced by communication overhead and I/O management is estimated to reach 9 μs. The curves in Fig. 13 must be shifted to the right side to account for this time delay and the maximum number of SMs/arm for the real-time CPU-based simulations now reduces to 160.

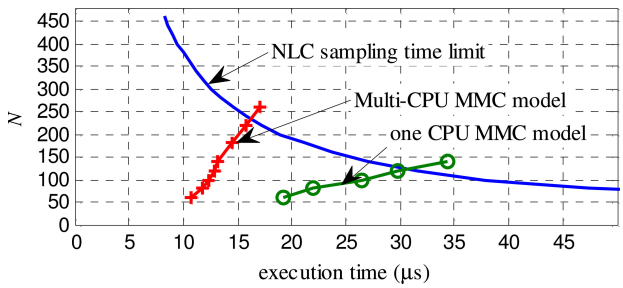

Fig. 13. N as a function of execution time for CPU-based models.

B. FPGA-Based Model

During the recent years, FPGA devices became an integral part of HIL simulators where they are used to provide access to digital and analog I/Os as well as for various computing purposes [23]. The density of modern FPGAs is nowadays sufficient to handle tens to hundreds of floating point operators, which enables them to offer tens GFLOPS of computing power. The FPGA used in this paper is the Virtex 6 LX240T [24] that comes with the ML605 board; the board is provided along with the OP5600.

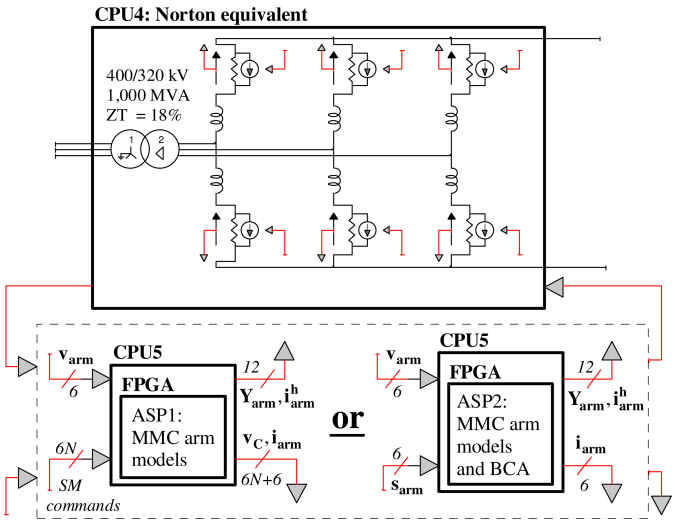

Fig. 14. Structure of the FPGA-based MMC model with ASP1 and ASP2.

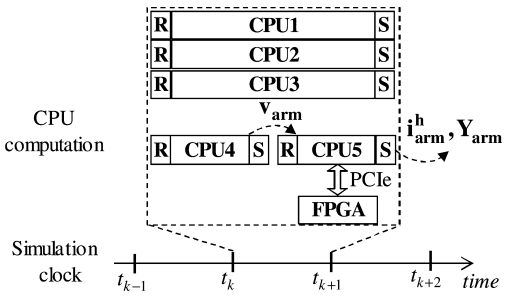

Fig. 15. Real-time computation for the FPGA-based MMC model.

Unlike other technologies (CPU, DSP or GPU), FPGA programming requires a complete design of a so-called application specific processor (ASP) to handle all the computations.

1) FPGA-Based MMC Model Implementation: The same algorithm of the arm model (see Fig. 6) implemented on CPU in [15] is used for the FPGA, except the implementation of SM blocked state code. The structure of the MMC model implemented on the FPGA is presented in Fig. 14. CPU4 contains the Norton equivalent circuit. At each simulation time point, CPU5 exchanges data with the FPGA through the PCIe protocol (see Fig. 15). Two versions of the ASP are implemented. ASP1 includes only six arm models. It receives $\mathbf{v_{arm}}$ and the SM commands, and sends back $\mathbf{i^h_{arm}}, \mathbf{R_{arm}}, \mathbf{v_C}$, and $\mathbf{i_{arm}}$ through the PCIe. ASP2 includes the six arm models and the BCA system. The communication load is now considerably reduced since $\mathbf{v_C}$ is not sent from the FPGA to CPU3 and the SM commands are not received from CPU3. Only $\mathbf{i_{arm}}$ and the number of SMs to be inserted ($\mathbf{s_{arm}}$) in each arm are exchanged between CPU3 and CPU5. Both ASPs are given the computing of arm Thevenin equivalent, which is converted to a Norton equivalent on CPU5. ASP2 is closer to a real HIL prototyping environment since the BCA acts like HIL through low latency FPGA I/O.

Fig. 16 illustrates the complete architecture of the ASP-based arm model. It is designed to handle up to 400 SMs/arm and to be easily configurable for other values of N. It consists of a sequencer for controlling the ASP, register files, and processing

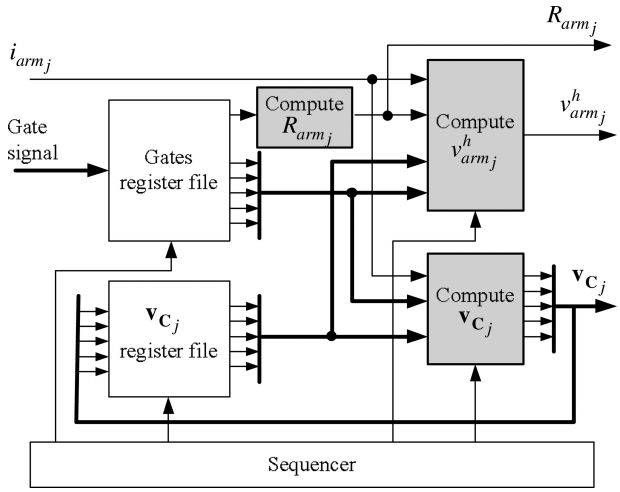

Fig. 16. ASP arm model implemented on the FPGA.

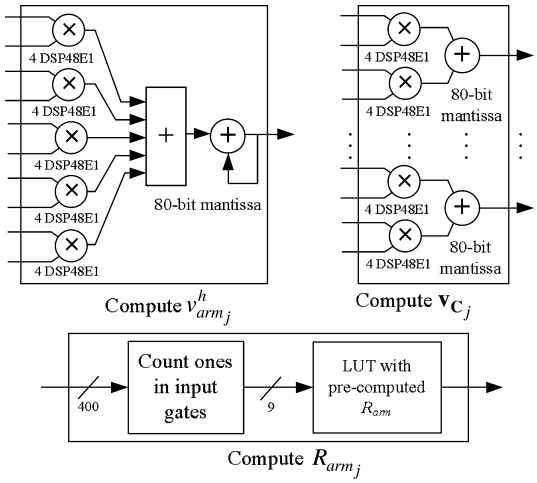

Fig. 17. Detailed view of the FPGA computing blocks.

elements. Register files are memory elements for storing $\mathbf{v_C}$ and SM commands. Each register file allows to read/write up to five values at the same time (parallel processing), leading to a maximum of 80 clock cycles for all values. The processing elements compute the components shown in Fig. 16 and require about 20 additional clock cycles once the reading task is completed.

The ASPs are able to handle up to six arm models by treating each arm model sequentially. This leads to a minimal time-step of 480 clock cycles (6×80), that is 2.4 μs for the three-phase MMC (the system clock frequency is 200 MHz). The latency of processing elements is not taken into account because the resulting $\mathbf{v_C}$ elements are rewritten to the register file while new values are processed for other arms. When the ASPs treat less than 400 SMs/arm, the processing elements are either idle or unused during the remaining clock cycles. The choices of treating arm models serially and reading five values at a time from the register files constitute a design tradeoff motivated by the fact that 2.4 μs is sufficient to conduct correctly a real-time simulation with 400 SMs/arm.

Fig. 17 presents the three processing elements that are identified by grayed out boxes in Fig. 16. The computed equations

TABLE I
DESIGN RESULTS FOR THE VIRTEX 6 LX240 T [24]

	ASP1	**ASP2**
Registers	33,302 (11%)	72,957 (24%)
LUTs	23,266 (15%)	39,702 (26%)
BRAM	92 (22%)	146 (35%)
DSP48E1	60 (7%)	84 (10%)

are described in the appendix. As stated in the introduction, an MMC may contain tens to hundreds of SMs/arm; thus all numerical values are represented with floating-point formats in order to allow a large dynamic range. A nonstandard floating-point format is used with intermediate precision (between simple and double), while all the additions (including accumulation) are carried out using an internal floating-point format called self-alignment format. This methodology has been introduced in [8], whereas the SAF is thoroughly discussed in [25]. It provides higher numerical accuracy while allowing the implementation of complex operators with very low latencies. More specifically, all the variables use a 35-bit signed mantissa, all the constants use a 44 signed mantissa, and all the additions are computed using an 80-bit extended mantissa [25]. Hence, each multiplication necessitates three clock cycles and four DSP blocs.

Table I presents the synthesis estimates of the ASPs for the Virtex 6 LX240 T. The two designs meet the timing constraint of 5 ns imposed by the 200-MHz system clock frequency. As it can be seen, ASP2 uses more resources than ASP1 because it also implements the BCA system. In the context where a physical controller is involved in the real-time simulation, the numbers of ASP1 hold with the real-time performances of ASP2.

2) Real-Time FPGA-Based Model Accuracy Verification: Before evaluating the real-time performances of the FPGA-MMC model, it is required to verify its accuracy. Only FPGA-ASP2 results are presented here, since it allows to verify simultaniously both arm models and the BCA system. The same three-phase-to-ground fault is reproduced. A MMC-401-level is simulated with a time-step of 9 μs. The FPGA-ASP2 results (green dotted line) are in real-time simulation, whereas the reference model (one-CPU MMC model) is simulated in an offline mode (blue solid line), since it cannot compute with these many SMs in real time (see Fig. 13).

Fig. 18 and Fig. 19 present simulation results. The differences between both model implementations are negligible and the relative errors vary between 0.3 and 3%. This is confirmed by other simulation cases.

3) Real-Time BCA System Accuracy Verification: In order to verify the implementation of the BCA, the capacitor voltages of SMs: SM_{1u_a}, SM_{400u_a}, $\mathrm{SM}_{1\ell_a}$, and $\mathrm{SM}_{400\ell_a}$ are presented in Fig. 20 under normal operation. Table II compares the average, minimum, and maximum values of capacitor voltages, and the numbers of switching-per-cycle for the same four SMs. A close match is achieved by the FPGA-ASP2 model. The relative errors for the capacitor voltages are below 0.1%. This performance is attributable to the high precision of the custom floating point operators. The average number of switching-per-cycle is within 0.5%, which proves the good accuracy of the FPGA-based BCA implementation.

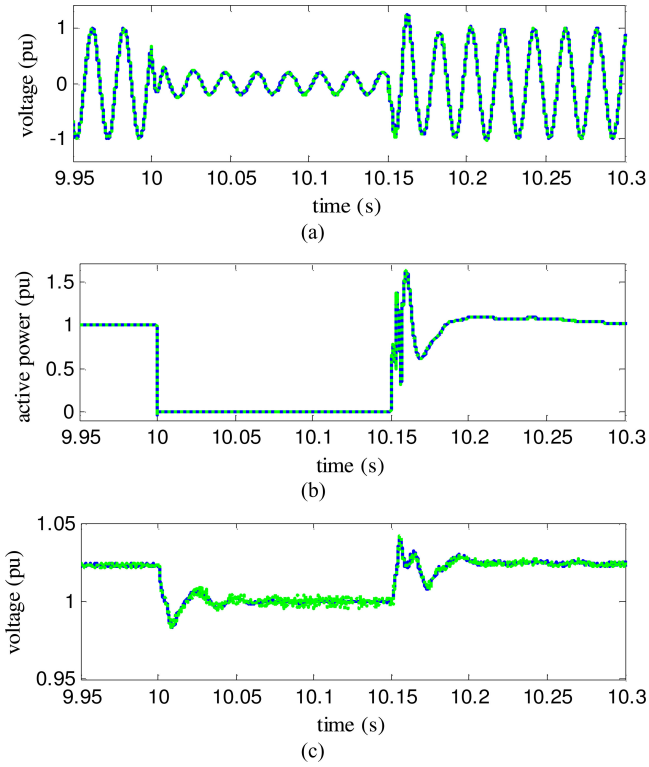

Fig. 18. MMC external variables. (a) Phase voltage: v_a. (b) Active power: P. (c) DC voltage: V_{dc}.

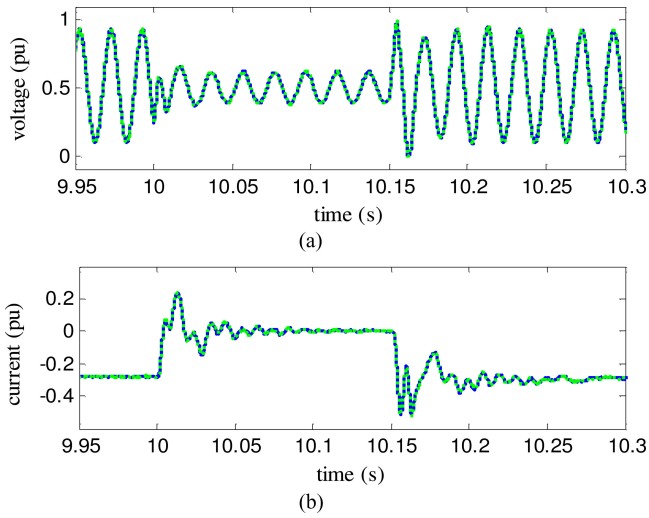

Fig. 19. MMC internal variables. (a) Arm voltage: v_{u_a}. (b) Circulating current: $(i_{u_a} + i_{l_a})/2$.

4) Real-Time Performance of the FPGA-Based Model: Real-time performances are evaluated in Fig. 21. For both ASPs, the FPGA runs at 2.4 μs; however, the execution times depicted in Fig. 21 include CPU4 and CPU5 since they are in series (see Fig. 14). The results indicate that the FPGA-ASP1 MMC model has a poor performance in comparison with the CPU-based model. This is explained by the important amount of data that has to be exchanged between the FPGA and the Fig. 14 CPU5 connected through the PCIe link ($2 \times 6\,N + 6$ values). This link

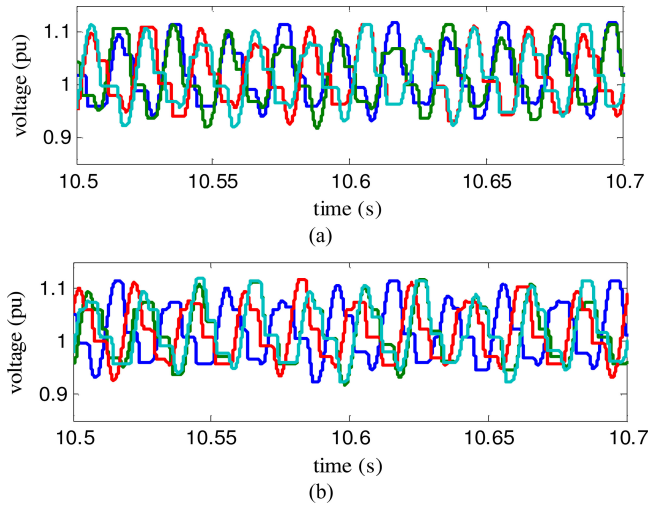

Fig. 20. Capacitor voltages of SM_{1u_a}, SM_{400u_a}, $\mathrm{SM}_{1\ell_a}$, and $\mathrm{SM}_{400\ell_a}$. (a) Capacitor voltages of the reference model (one-CPU model). (b) Capacitor voltages of the FPGA-ASP2 model.

TABLE II
ACCURACY VERIFICATION OF THE BCA IMPLEMENTED ON THE FPGA

		One-CPU MMC	FPGA-ASP2	Relative error (%)
Capacitor voltages (pu)	Max	1.1188	1.1187	0.0064
	Average	1.0167	1.0176	0.0847
	Min	0.9138	0.9139	0.0215
SM switching per cycle	Max	8	8	0
	Average	4.9935	4.9685	0.4998
	Min	1.25	1.25	0

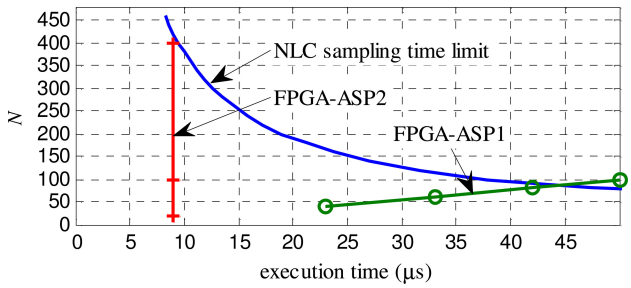

Fig. 21. N as a function of execution time for FPGA configurations.

has a latency that is higher than that of a shared-memory-based communication. However, this situation is less realistic in regard of the HIL simulation context where the real controller is physically connected to the simulator by means of the FPGA.

The real-time performance of the FPGA-ASP2 model is the best among those reported in this paper. It is able to achieve an execution time of 9 μs. Moreover, it does not depend on the number of SMs (see setup Fig. 14). The FPGA-ASP2 setup is more realistic within the context of HIL simulation, where the gating signals are provided by a physical controller to the FPGA through its low latency I/Os.

VI. CONCLUSION

This paper presented a feasability study for the real-time simulation of an MMC model varying from 61 to 401 levels. It has been shown that the CPU-based MMC model is limited to 231 levels. This limit is reduced to 161 levels for HIL simulations due to latencies in communications and I/Os.

This paper also presents two different FPGA-based MMC model setups. In the first setup (ASP1), the high amount of data exchanged between arm models and the control system does not allow to achieve acceptable performance. In the second setup (ASP2), the BCA is implemented on the FPGA and the lowest execution time of 9 μs is achieved. Moreover, the FPGA based MMC arm models run at 2.4 μs. These execution times do not depend on the number of MMC levels. No additional latency is expected for HIL simulation, since the arm models are coded on the FPGA.

For MMC topologies with more than 400 SMs/arm, the presented FPGA setup has sufficient supplementary resources to handle more powerful ASPs. The computing power of such ASPs is a design tradeoff that depends on the number of SMs per arm and the targetted simulation time-step. In fact, the actual bottleneck is in communication time between the FPGA and the remaining CPU-based network model.

APPENDIX

The processing elements $v^h_{\mathrm{arm}_j}$ and $\mathbf{v_C}_j$ shown in Fig. 16 are found from their equations [15]. $v^h_{\mathrm{arm}_j}$ (jth arm) is given by

$$v^h_{\mathrm{arm}_j}(t) = R_{\mathrm{arm}_j} i_{\mathrm{arm}_j}(t) + \sum_{i=1}^{N} \alpha_{ij} v_{C_{ij}}(t) \tag{4}$$

where α_{ij} depends on the gating signal of the i*th* SM. The capacitor voltage for the following time-point is given by

$$v_{C_{ij}}(t + \Delta t) = R_{\mathrm{eq}_{ij}} i_{\mathrm{arm}_j}(t) + \beta_{ij} v_{C_{ij}}(t) \tag{5}$$

where $R_{\mathrm{eq}_{ij}}$ and β_{ij} also depend upon the gating. Finally, R_{arm_j} is found from the gating signals as follows. The blocking state not being considered in this paper, each SM is either ON or OFF. R_{arm_j} depends on the number of SMs that are ON and can be obtained from a table that is addressed using the ON-state SM number, as shown in Fig. 17

The ASP computing power can be estimated as follows: $v^h_{\mathrm{arm}_j}$ involves five multiplications and five additions, for a total of ten floating-point operations; $\mathbf{v_C}_j$ necessitates 10 multiplications and 5 additions, for a total of 15 floating-point operations; the ASP runs at 200 MHz, which gives a computing power of $(15 + 10) \times 0.2 = 5$ GFLOPS. For a given MMC with 400 SMs, the ASPs run at peak performance.

REFERENCES

[1] B. Gemmell, J. Dorn, D. Retzmann, and D. Soerangr, "Prospects of multilevel VSC technologies for power transmission," in *Proc. IEEE Transmiss. Distrib. Conf. Exp.*, Milpitas, CA, USA, Apr. 2008, pp. 1–16.
[2] N. Flourentzou, V. G. Agelidis, and G. D. Demetriades, "VSC-based HVDC power transmission systems: An overview," *IEEE Trans. Power Electron.*, vol. 24, no. 3, pp. 592–602, Mar. 2009.
[3] A. Lesnicar and R. Marquardt, "An innovative modular multilevel converter topology suitable for a wide power range," presented at the IEEE Power Tech. Conf., Bologna, Italy, Jun. 2003.
[4] B. R. Andersen, L. Xu, and K. T. G. Wong, "Topologies for VSC transmission," in *Proc. 7th Int. Conf. AC–DC Power Transmiss.*, London, U.K., Nov. 2001, pp. 298–304.
[5] S. Allebrod, R. Hamerski, and R. Marquardt, "New transformerless, scalable modular multilevel converters for HVDC-transmission," in *Proc. Power Electron. Spec. Conf.*, Jun. 15–19, 2008, pp. 174–179.

[6] R. S. Whitehouse, C. D. M. Oates, J. Maneiro, and N. M. MacLeod, "A new simulator laboratory for research and development of VSC HVDC topologies and control algorithms," in *Proc. 9th IET Int. Conf. AC DC Power Trans.*, Oct. 19–21, 2010, pp. 1–5.
[7] M. Matar and R. Iravani, "FPGA implementation of the power electronic converter model for real-time simulation of electromagnetic transients," *IEEE Trans. Power Del.*, vol. 25, no. 2, pp. 852–860, Apr. 2010.
[8] T. Ould-Bachir, C. Dufour, J. Bélanger, J. Mahseredjian, and J. P. David, "A fully automated reconfigurable calculation engine dedicated to the real-time simulation of high switching frequency power electronic circuits," *Math. Comput. Simul.*, vol. 91, pp. 167–177, May 2013.
[9] S. P. Teeuwsen, "Simplified dynamic model of a voltage-sourced converter with modular multilevel converter design," in *Proc. IEEE Power Syst. Conf. Exp.*, Seattle, WA, USA, Mar. 2009, pp. 1–6.
[10] J. Peralta, H. Saad, S. Dennetière, J. Mahseredjian, and S. Nguefeu, "Detailed and Averaged Models for a 401-level MMC-HVDC system," *IEEE Trans. Power Del.*, vol. 27, no. 3, pp. 1501–1508, Jul. 2012.
[11] H. Jin, "Behavior-mode simulation of power electronic circuits," *IEEE Trans. Power Electron.*, vol. 12, no. 3, pp. 443–452, May 1997.
[12] 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. Power Del.*, vol. 26, no. 1, pp. 316–324, Jan. 2011.
[13] P. Le-Huy, P. Giroux, and J.-C. Soumagne, "Real-time simulation of modular multilevel converters for network integration studies," presented at the Int. Conf. Power Systems Transients, Delft, The Netherlands, Jun. 14–17, 2011.
[14] L.-A. Grégoire, Wei Li, J. Bélanger, and L Snider, "Validation of a 60-level modular multilevel converter model— Overview of offline and real-time HIL testing and results," presented at the Int. Conf. Power Systems Transients, Delft, The Netherlands, Jul. 14, 2011.
[15] H. Saad, C. Dufour, J. Mahseredjian, S. Dennetière, and S. Nguefeu, "Real time simulation of MMCs using the state-space nodal approach," presented at the Int. Conf. Power Systems Transients, Vancouver, BC, Canada, Jul. 18–20, 2013.
[16] M. Hagiwara and H. Akagi, "PWM control and experiment of modular multilevel converter," in *Proc. IEEE Power Electron. Spec. Conf.*, Tokyo, Japan, Jun. 2008, pp. 154–161.
[17] L. G. Franquelo, J. Rodriguez, I. J Leon,, S. Kouro, R. Portillo, and M. A. M. Prats, "The age of multilevel converters arrives," *IEEE Ind. Electron. Mag.*, vol. 2, no. 2, pp. 28–39, Jun. 2008.
[18] A. Antonopoulos, L. Angquist, and H. P. Nee, "On dynamics and voltage control of the modular multilevel converter," in *Proc. 13th Eur. Conf. Power Electron. Appl.*, Barcelona, Spain, Sep. 2009, pp. 1–10.
[19] H. Peng, M. Hagiwara, and H. Akagi, "Modeling and analysis of switching-ripple voltage on the DC link between a diode rectifier and a modular multilevel cascade inverter (MMCI)," *IEEE Trans. Power Electron.*, vol. 28, no. 1, pp. 75–84, Jan. 2013.
[20] T. Qingrui and Xu. Zheng, "Impact of sampling frequency on harmonic distortion for modular multilevel converter," *IEEE Trans. Power Del.*, vol. 26, no. 1, pp. 298–306, Jan. 2011.
[21] C. Dufour, J. Mahseredjian, and J. Belanger, "A combined state-space nodal method for the simulation of power system transients," *IEEE Trans. Power Del.*, vol. 26, no. 2, pp. 928–935, 2011.
[22] eMEGAsim. (2013). [Online]. Available: http://www.opal-rt.com/product/emegasim
[23] T. Ould-Bachir, C. Dufour, J. Belanger, J. Mahseredjian, and J. P. David, "Effective floating-point calculation engines intended for the FPGA-based HIL simulation," in *Proc. IEEE Int. Symp. Ind. Electron.*, May 2012, pp. 1363–1368.
[24] *Xilinx Data sheet DS150 v2.4. Virtex 6 family overview.* Xilinx, San Jose, CA, USA, Jan. 2012.
[25] T. Ould-Bachir and J. P. David, "Self-alignment schemes for the implementation of addition-related floating-point operators," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 6, no. 1, Article 1, pp. 1–21, May 2013.

Hani Saad (S'07) received the B.Sc. degree in electrical engineering from École Polytechnique de Montréal, Montreal, QC, Canada, in 2007. He is currently working toward the Ph.D. degree in electrical engineering at the École Polytechnique de Montréal, Montreal.

From 2008 to 2010, he was with Techimp and in the Laboratory of Materials Engineering and High Voltages (LIMAT) of the University of Bologna on research and development activities related to partial-discharge diagnostics in power systems.

Tarek Ould-Bachir (M'08) received the B.Eng., M.A.Sc., and Ph.D. degrees in electrical engineering from École Polytechnique de Montréal, Montreal, QC, Canada, in 2005, 2008, and 2013, respectively.

Since 2005, he has been lecturing undergraduate courses at École Polytechnique de Montréal. From 2007 to 2009, he was an FPGA Application Specialist with Opal-RT Technologies. His research interests include real-time simulation, digital arithmetic, and stochastic computing.

Jean Mahseredjian (F'13) received the M.A.Sc. and Ph.D. degrees from École Polytechnique de Montréal, Montreal, QC, Canada, in 1985 and 1991, respectively.

From 1987 to 2004, he was with IREQ (Hydro-Québec) on research and development activities related to the simulation and analysis of electromagnetic transients. In December 2004, he joined the faculty of electrical engineering at École Polytechnique de Montréal.

Christian Dufour (M'96) received the Ph.D. degree from Laval University, QC, Canada, in 2000.

He joined Opal-RT Technologies, Montreal, QC, Canada, in 1999 where he is the Lead Researcher in electric system simulation software for RT-LAB. Before joining Opal-RT, he worked on the development teams of Hydro-Québec HYPERSIM real-time simulator as well as the MathWorks SimPowerSystems blockset. His current research interests include algorithmic solutions for the real-time simulation of power systems and motor drives.

Sébastien Dennetière (M'04) received the Graduation degree from École Supérieure d'Electricité (Supélec), Paris, France, in 2002. He received the M.A.Sc. degree from École Polytechnique de Montréal, Montreal, QC, Canada, in 2003.

From 2002 to 2004, he worked at IREQ (Hydro-Québec) on research and development activities related to the simulation and analysis of electromagnetic transients. From 2004 to 2009, he was with the Research Center of EDF in the field of insulation coordination and power system simulations. In January 2010, he joined the French TSO RTE (Réseau de Transport d'Electricité), Paris, France, where he is currently involved in power system simulation.

Samuel Nguefeu (M'04) received the Graduation degree from École Supérieure d'Electricité (Supélec), Paris, France, in 1991. He received the M.A.Sc. and Ph.D. degrees from Université Pierre et Marie (Paris VI), Paris, France, in 1991 and 1993, respectively.

He worked as a Consultant for 2 years before joining THOMSON in 1996. From 1999 to 2005, he worked at EDF R&D in Power Systems and Power Electronics. In November 2005, he joined the French TSO (RTE), Paris, France, where he is currently involved in FACTS and HVDC projects.