

Exact Nonlinear Micromodeling for Fine-Grained Parallel EMT Simulation of MTDC Grid Interaction With Wind Farm

Ning Lin , Student Member, IEEE, and Venkata Dinavahi , Senior Member, IEEE

Abstract—Detailed high-order models of the insulated-gate bipolar transistor (IGBT) and the diode are rarely included in power converters for large-scale system-level electromagnetic transient (EMT) simulation on the CPU, due to the nonlinear characteristics albeit they are more accurate. The massively parallel architecture of the graphics processing unit (GPU) enables a lower computational burden by avoiding the computation of complex devices repetitively in a sequential manner and thus is utilized in this paper to simulate the wind farm-integrated multiterminal dc (MTdc) grid based on the modular multilevel converter (MMC). Fine-grained circuit partitioning is proposed so that the nonlinear switching elements are physically separated with the smallest circuit unit. By implementing these subsystems with the same attributes as a GPU program and computing it in a massively parallel manner, it is demonstrated that the GPU is able to achieve a significant speedup over multicore CPUs and its computation time incremental is much smaller when the MMC level scales up. The improved insight and accuracy of the proposed modeling methodology and the designed GPU program are validated at the system- and device-level by off-line commercial simulation tools.

Index Terms—Doubly fed induction generator (DFIG), electromagnetic transients, graphics processors, high voltage direct current (HVdc), massively parallel processing, modular multilevel converter (MMC), multiterminal dc (MTdc), nonlinear IGBT model, parallel algorithm, wind farm.

I. INTRODUCTION

HIGH voltage direct current (HVdc) transmission based on the modular multilevel converter (MMC) has shown its potential in expanding into a multiterminal dc (MTdc) grid by interconnecting multiple converter stations [1]. Although a few projects have been under operation, new system configurations [2]–[4], novel control, and protection strategies [5]–[7] keep springing up, and the prime means of studying them is electromagnetic transient (EMT) simulation, where tremendous

efforts have been paid to improve the efficiency while retaining a reasonable accuracy.

The two-state switch model (TSSM) with fixed ON- and OFF-state resistances for semiconductor switches is the main model that current mainstream system-level EMT-type simulation tools such as PSCAD/EMTdc adopt [8]. However, it is seldom seen in MMC-based HVdc simulation where hundreds of insulated-gate bipolar transistors (IGBTs) and their antiparallel diodes constitute a huge circuit that is too burdensome for CPUs to solve. Therefore, aggregated models such as the detailed equivalent model and the averaged value model and their variants are proposed and implemented in EMT-type solvers [9]–[11], all of which are unable to surpass the TSSM in terms of accuracy. Furthermore, even the TSSM is deemed as an ideal model which falls short of revealing transient behavior of the IGBT, e.g., the inaccessibility of current overshoots during switching in the EMT simulators could lead to an underestimation of its actual stress and subsequently the inappropriate selection of a device type with inadequate capacity; the MMC power loss is another aspect that needs attention, where currently some simple switch models are applied for loss estimation [12], [13]. Thus, it is imperative to include nonlinear device-level switch models in EMT solvers for large system simulation to gain adequate insight [14].

On the contrary, for small-scale converter design, it is practical to include device-level IGBT and diode models that are experimentally validated in simulation by platforms such as SaberRD and PSpice [15]–[17]. Reproducing sufficient accuracy in describing the static and dynamic behaviors of a real switch is the noticeable drawback of CPU's inefficiency in solving the converter even with a few such devices, let alone megawatt power converters such as the MMC. Moreover, with numerous circuit nodes linking linear and nonlinear components, the simulation is commonly subjected to frequent termination due to numerical divergence. All the aforementioned facts account for the absence of nonlinear IGBT and diode models from the MMC topology for MTdc grid simulation.

Circuit partitioning brings a prominent speedup to EMT simulation by calculating a number of small circuits [18], rather than the original circuit that corresponds to an extremely large admittance matrix. Thus, it is one of the fundamental methods in accelerating the simulation of MTdc grid with micromodeling of MMCs. Nevertheless, due to the sequential implementation

Manuscript received April 6, 2018; revised June 18, 2018; accepted July 12, 2018. Date of publication August 6, 2018; date of current version March 29, 2019. This work was supported by Natural Science and Engineering Research Council of Canada. (Corresponding author: Ning Lin.)

The authors are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada (e-mail: ning3@ualberta.ca; dinavahi@ualberta.ca).

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/TIE.2018.2860566

of these subcircuits by the CPU, the simulation still takes a long time even for a small fraction of duration.

The graphics processing unit (GPU) has been employed previously for electromagnetic transient simulation of power systems and power electronic circuits [19], [20]. Therefore, in this paper, its massive parallelism is utilized to expedite the simulation of MTdc grid involving microlevel details. The submodule (SM) containing nonlinear behavioral IGBT/diode pair contributes to the main computational burden within the MMC. Thus, coupled voltage–current (V – I) sources are used for creating a fine-grained network from the original system structure by separating all MMC SMs, whose similarity enables writing them as one global function by the GPU programming language CUDA C [21] and parallel execution by multiple computational blocks and threads. Consequently, the GPU is able to gain a remarkable speedup over CPUs.

This paper is organized as follows. Section II introduces system-level modeling of the MTdc grid. The IGBT/diode nonlinear behavioral model and the GPU program design of an MMC is discussed in Section III. GPU implementation results are presented and validated in Section IV, and Section V provides conclusions.

II. WIND FARM-INTEGRATED MTDC GRID

Fig. 1 shows the CIGRÉ B4 dc grid [1] integrated with offshore wind farms (OWFs). It comprises 3 dc systems (DCSs) and 11 ac/dc terminals. The onshore converter stations connecting with OWF1–5 are rectifiers so that the energy could be transmitted to inland inverters.

A. Induction Machine Model

The induction machine (IM) is the core part that converts the wind's kinetic energy into electricity. It is based on the following state-space equations [22]:

$$\dot{\Phi} = \mathbf{A}\Phi + \mathbf{B}\mathbf{U} \quad (1)$$

$$\mathbf{I} = \mathbf{C}\Phi \quad (2)$$

where Φ , \mathbf{I} , and \mathbf{U} are vectors of fluxes, currents, and excitations of the DFIG in the α – β frame, respectively. Since their elements are arranged in the same sequence, they can be uniformly denoted by a symbolic vector \mathbf{X} as

$$\mathbf{X} = [X_{\alpha s}, X_{\beta s}, X_{\alpha r}, X_{\beta r}]^T. \quad (3)$$

Here, the subscript α and β represent the α – β frame, and s and r indicate variables belonging to either the stator or rotor, respectively. The input matrix \mathbf{B} is a 4×4 identity matrix, while the state and the output matrices are given as

$$\mathbf{A} = \begin{bmatrix} \frac{-R_s L_r}{L_s L_r - L_m^2}, & 0, & \frac{R_s L_m}{L_s L_r - L_m^2}, & 0 \\ 0, & \frac{-R_s L_r}{L_s L_r - L_m^2}, & 0, & \frac{R_s L_m}{L_s L_r - L_m^2} \\ \frac{R_r L_m}{L_s L_r - L_m^2}, & 0, & \frac{-R_r L_s}{L_s L_r - L_m^2}, & -\omega_r \\ 0, & \frac{R_r L_m}{L_s L_r - L_m^2}, & \omega_r, & \frac{-R_r L_s}{L_s L_r - L_m^2} \end{bmatrix} \quad (4)$$

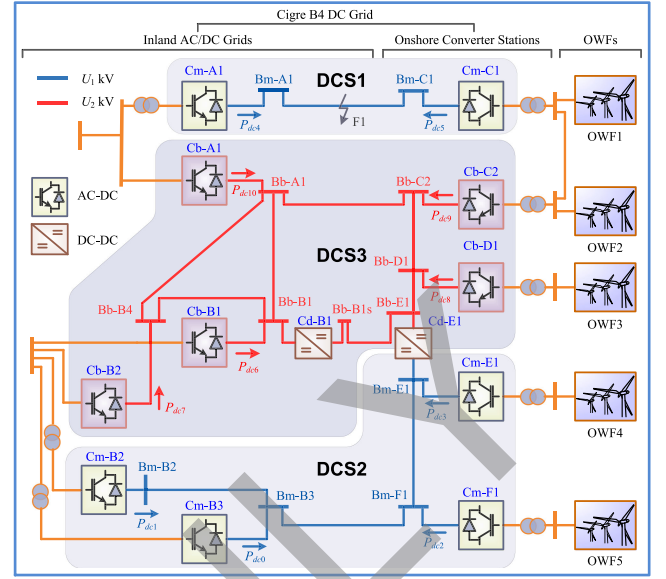


Fig. 1. The CIGRÉ B4 dc grid integrated with OWFs.

$$\mathbf{C} = \begin{bmatrix} \frac{L_r}{L_s L_r - L_m^2}, & 0, & \frac{-L_m}{L_s L_r - L_m^2}, & 0 \\ 0, & \frac{L_r}{L_s L_r - L_m^2}, & 0, & \frac{-L_m}{L_s L_r - L_m^2} \\ \frac{-L_m}{L_s L_r - L_m^2}, & 0, & \frac{L_s}{L_s L_r - L_m^2}, & 0 \\ 0, & \frac{-L_m}{L_s L_r - L_m^2}, & 0, & \frac{L_s}{L_s L_r - L_m^2} \end{bmatrix}. \quad (5)$$

where L_s , L_r , and L_m are the stator, rotor, and magnetizing inductances, respectively; R_s and R_r are the stator and rotor resistances, respectively; and ω_r is the electrical rotor velocity.

The electromagnetic torque is calculated following the solution of the space-state equations, as

$$T_e = 1.5 P_p (\Phi_{\alpha s} I_{\beta s} - \Phi_{\beta s} I_{\alpha s}) \quad (6)$$

where P_p is the number of pole pairs and ω_r is subsequently calculated by

$$\omega_r = \int \frac{P}{J} (T_e - T_m) dt \quad (7)$$

where J is the inertia and T_m is the mechanical torque.

B. Aggregated Wind Farm EMT Model

The rectifier collects wind energy and provides a stable ac voltage to an array of doubly fed induction generators (DFIGs) which may be located in regions where ac grid is not available. As shown in the d – q frame-based controller in Fig. 2, the ac voltage reference in the d -axis V_{gd}^* is set for the actual output voltage to follow, and phase-shift control (PSC) comprising of the averaging (AVG) and balancing (BC) algorithms is adopted for regulating the MMC SM capacitor voltages [23]. The configuration of the rectifier side is given in Fig. 3(a). The wind turbine model converts wind speed into torque and feeds it into the IM, which generates electric power under vector control [24]. The OWF is represented by aggregated DFIGs since the

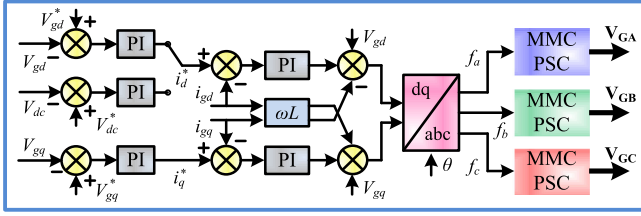


Fig. 2. MMC-based converter station controller.

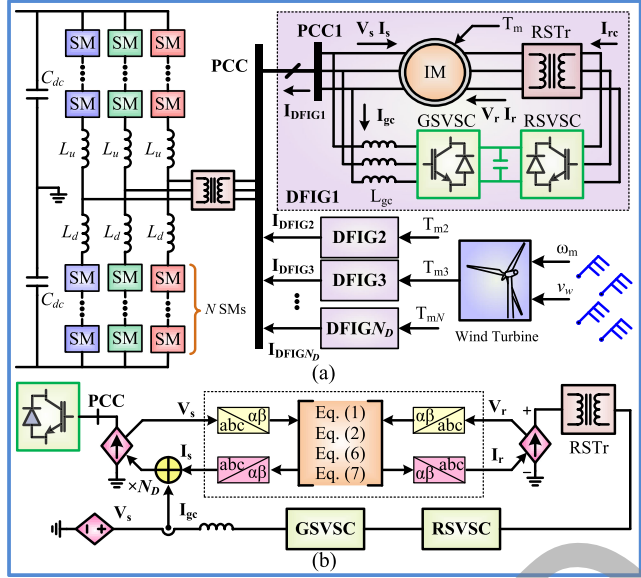


Fig. 3. OWF integration into MTdc grid: (a) DFIG array connected with MMC and (b) rectifier-side EMT model with aggregated wind farms.

focus is on GPU simulation of power converters with nonlinear device-level details for system study.

From the perspective of circuit analysis, the rectifier side contains a large number of nodes. Moreover, the IM model cannot be solved along with its surrounding parts by a matrix equation. Thus, twofold $V-I$ source couplings are introduced for interfacing the IM to external circuits, as shown in Fig. 3(b). The voltage sources should be on the induction machine's side since the input of (1) is three-phase voltage, and the current sources are solved in conjunction with the remaining parts.

The aggregation of all DFIGs in an OWF introduces the other coupling at the point of common coupling (PCC). The voltage source is placed on the DFIG side, and on the ac side of the rectifier is the three-phase current-controlled current sources, with a scaling factor N_D representing the number of DFIGs in a wind farm.

C. Circuit Partitioning for Simulation Speedup

The basic configuration of a three-phase $(N + 1)$ -level MMC is given in Fig. 3(a), where the N -cascaded SMs in each arm would pose a severe computational challenge for efficient simulation on processors if not simplified or separated, particularly when the IGBT/diode nonlinear models are adopted. The relatively low frequency of arm currents compared with that of

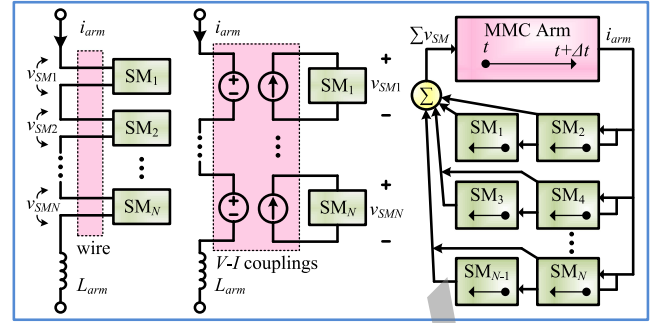
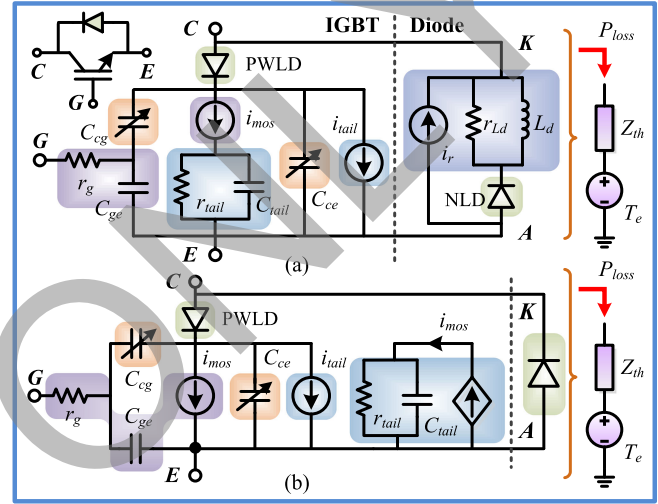
Fig. 4. MMC SM partitioning by $V-I$ couplings.

Fig. 5. Nonlinear IGBT/diode electrothermal model: (a) Original behavioral model and (b) improved model.

EMT computation means that between two neighboring time steps, the currents can be deemed as constant; thus, they satisfy the principle of circuit partitioning by $V-I$ coupling which split all SMs from their corresponding arms to constitute the minimum separable subcircuits for alleviating the computational burden and improving numerical stability, as shown in Fig. 4. On the left, the MMC arm collects all SM voltages, while at the same time the SMs also receive the arm current as their inputs. Then, the two types of circuits conduct computation without interfering with each other until completion, when a new round of information exchange takes place and the time step moves forward by Δt .

III. NONLINEAR BEHAVIORAL IGBT/DIODE ELECTROTHERMAL MODEL-BASED MMC

A. IGBT and Diode Pair EMT Model

The nonlinear behavioral model (NBM) of the IGBT and its antiparallel diode is taken as an example for the demonstration of sophisticated device-level switches involved in system-level simulation on the GPU. As the equivalent circuit of the pair shown in Fig. 5(a), the IGBT contains five nodes while the diode has three nodes [25], [26], but since the latter shares its anode and cathode with the emitter and collector of the former, the original model has totally six nodes.

1) IGBT Model: The MOSFET behavior represented by the current source i_{mos} is the pivotal part. Together with capacitors C_{ge} , C_{cg} and the inherent gate resistance r_g , it reflects static and dynamic performance other than the tailing current by the following three sections corresponding to the OFF state, the ON state, and the transient state, respectively:

$$i_{\text{mos}} = \begin{cases} 0, & (v_{\text{cge}} < V_t) \vee (v_d \leq 0) \\ a_2 \cdot v_d^{(z+1)} - b_2 \cdot v_d^{(z+2)}, & v_d < (y \cdot (v_{\text{cge}} - V_t))^{\frac{1}{x}} \\ \frac{(v_{\text{cge}} - V_t)^2}{a_1 + b_1 \cdot (v_{\text{cge}} - V_t)}, & (\text{others}) \end{cases} \quad (8)$$

where a_1, b_1, a_2, b_2, x, y , and z are coefficients, V_t is the constant channel threshold voltage, and v_d and v_{cge} are the voltage over i_{mos} and C_{ge} . As a consequence, discretization of the component yields conductance G_{mosvd} and transconductance G_{mosvcge} derived by taking partial derivatives with respect to v_d and v_{cge} , e.g., under ON state

$$G_{\text{mosvd}} = \frac{\partial i_{\text{mos}}}{\partial v_d} = a_2(z+1) \cdot v_d^z - b_2(z+2) \cdot v_d^{z+1} \quad (9)$$

and under the transient stage

$$G_{\text{mosvcge}} = \frac{2(v_{\text{cge}} - V_t)}{a_1 + b_1(v_{\text{cge}} - V_t)} - \frac{b_1(v_{\text{cge}} - V_t)^2}{(a_1 + b_1(v_{\text{cge}} - V_t))^2}. \quad (10)$$

The Norton equivalent current I_{moseq} can be expressed by the following combination:

$$I_{\text{moseq}} = i_{\text{mos}} - G_{\text{mosvd}} \cdot v_d - G_{\text{mosvcge}} \cdot v_{\text{cge}}. \quad (11)$$

The IGBT tailing current phenomenon concentrates on $r_{\text{tail}}-C_{\text{tail}}$ pair and the current source i_{tail}

$$i_{\text{tail}} = \begin{cases} 0 & (v_{\text{tail}}/r_{\text{tail}} di_{\text{mos}}) \\ \left(\frac{v_{\text{tail}}}{r_{\text{tail}}} - i_{\text{mos}}\right) \cdot i_{\text{rat}} & (v_{\text{tail}}/r_{\text{tail}} > i_{\text{mos}}) \end{cases} \quad (12)$$

The reactive elements such as the capacitor and inductor expressed in ordinary differential equations should be discretized for EMT simulation. Implicit one-step integration methods are used: the second-order Trapezoidal rule and Backward Euler method are adopted in discretizing system-level components and the nonlinear IGBT/diode model, respectively, since the small simulation time step ensures their accuracy. Other integration methods with higher orders are not used due to a heavier computation burden although they are also applicable.

2) Diode Model: The diode is essential to the normal operation of an MMC since it provides a conduction path when the IGBT is turned OFF. Its behavioral model contains two parts: the nonlinear diode symbolized by *NLD* exhibits complex static characteristics, and the remaining three components are in charge of the reverse recovery feature. The basic I - V relationship is realized by the exponential function

$$I_d = I_s \cdot \left(e^{\frac{V_j}{V_b}} - 1 \right) \quad (13)$$

where V_j is the static junction voltage, I_s is the leakage current, and V_b the junction barrier potential. Discretization and linearization toward this nonlinear component are carried out so

that a discrete-time Norton equivalent circuit becomes available, derived by

$$G_j = \frac{\partial I_d}{\partial V_j} = \frac{I_s}{V_b} e^{\frac{V_j}{V_b}} \quad (14)$$

$$I_{\text{jeq}} = I_d - G_j \cdot V_j \quad (15)$$

where G_j and I_{jeq} represent conductance and the equivalent current source.

The Backward Euler integration is sufficient for discretization of linear parts since the simulation time step is small. Thus, the discretized model of the conductor is expressed as

$$G_{L_d} = \frac{L_d}{\Delta t} \quad (16)$$

$$I_{L_d \text{eq}}(t) = i_{L_d}(t - \Delta t) \quad (17)$$

and the current source i_r is controlled by the voltage over the $r_{L_d}-L_d$ pair with an amplification coefficient of K_r .

3) Improved IGBT/Diode Model: Small-scale power converter simulation with the above model is practical as the CPU can solve the corresponding matrix with convergent results. However, in the MMC, even with partitioned configuration, the ten nodes in one SM is still too burdensome to compute; moreover, MATLAB simulation revealed that the $r_{\text{tail}}-C_{\text{tail}}$ pair in the IGBT is the main source that contributes to numerical divergence. Thus, the model is modified, as shown in Fig. 5(b). As the $r_{\text{tail}}-C_{\text{tail}}$ pair only impacts the tail current i_{tail} , it can be removed from its initial position and constitute an independent circuit. In EMT simulation, after obtaining i_{mos} , the value is used to compute the nodal voltage of the RC pair, which further yields the tail current. The freewheeling diode can either be kept unchanged or, under certain circumstances its reverse recovery could be neglected, retaining only the *NLD* for higher efficiency.

In this case, discretization of the controlled current source i_{tail} becomes unnecessary, since its value can be directly calculated by (12), which shortens the computation time. Meanwhile, in the partitioned SM, there would be only eight nodes; the omission of two nodes not only improves simulation speed but also reduces the chance of numerical divergence.

The electrothermal network is necessary for micromodeling of the switch. The power dissipation is sent to the network which can be represented by an RC network denoted by Z_{th} , and T_e is the ambient temperature.

B. IGBT/Diode Grouping

The MMC SM unit may contain a number of IGBT/diode pairs, e.g., a fundamental half-bridge SM (HBSM) contains two switches, and in case the rating of a single IGBT is not enough, one switch symbol may be comprised of a few parallel IGBTs. Consequently, the number of nodes in the split SM rises dramatically and the simulation will be further slowed down. However, noticing that during operation, they are well-balanced and the internal nodes have the same potential, a scaling factor m is introduced to indicate their exact number. Then, as indicated in Fig. 6(a), the basic admittance matrix for the HBSM still has

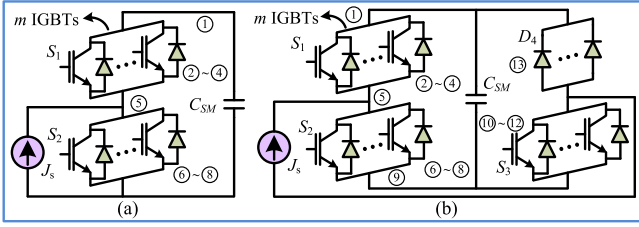


Fig. 6. Partitioned SM nonlinear behavioral model: (a) HBSM and (b) FBSM.

a dimension of 8, written as a combination of three parts

$$\mathbf{G}_{SM} = \begin{bmatrix} G_C & \mathbf{0}_{1 \times 7} \\ \mathbf{0}_{7 \times 1} & \mathbf{0}_{7 \times 7} \end{bmatrix} + \begin{bmatrix} m \cdot \mathbf{G}_{S5 \times 5} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{4 \times 4} & \mathbf{0}_{4 \times 4} \\ \mathbf{0}_{4 \times 4} & m \cdot \begin{bmatrix} G_{S11} & \cdots & G_{S14} \\ \vdots & \ddots & \vdots \\ G_{S41} & \cdots & G_{S44} \end{bmatrix} \end{bmatrix} \quad (18)$$

where the element G_C represents the conductance of SM capacitor; the second matrix lists all elements in the five-node upper switch, while its lower counterpart is placed in the third matrix, which only contains 16 elements after the fifth node is naturally grounded. Similarly, the current contribution vector can be expressed by

$$\mathbf{J}_{SM} = [I_{hisC} \ 0 \ 0 \ 0 \ J_s \ 0 \ 0 \ 0] + m \cdot [J_{S1} \ J_{S2} \ J_{S3} \ J_{S4} \ J_{S5} \ 0 \ 0 \ 0] + m \cdot [0 \ 0 \ 0 \ 0 \ J_{S1} \ J_{S2} \ J_{S3} \ J_{S4}] \quad (19)$$

where I_{hisC} and J_s are the capacitor's history current and arm current, respectively. The SM nodal voltages are subsequently obtained by

$$\mathbf{U}_{SM} = \mathbf{G}_{SM}^{-1} \cdot \mathbf{J}_{SM}. \quad (20)$$

The full-bridge SM (FBSM) is another topology used in MMC-based HVdc transmission for dc-side fault ride-through. It originally contains 15 nodes; however, considering that during normal operation the fourth IGBT is constantly under OFF state, it can be omitted, and the new FBSM in Fig. 6(b) has 13 nodes. Its admittance and current contribution matrices can be acquired in a similar style and the nodal voltages are also calculated by (20).

C. Fine-Grained MMC GPU Kernel Design

The circuit partition creates a substantial number of components, and those with the same attribute are written as one GPU program. The MMC is the most complicated part in the MTdc system for the existence of several hierarchies, e.g., one converter contains three legs, each of which has two arms, and in an arm, there are a number of SMs. A proper GPU program design must lead to a high level of parallelism, meaning components in the lower hierarchy have a higher priority in the design, e.g.,

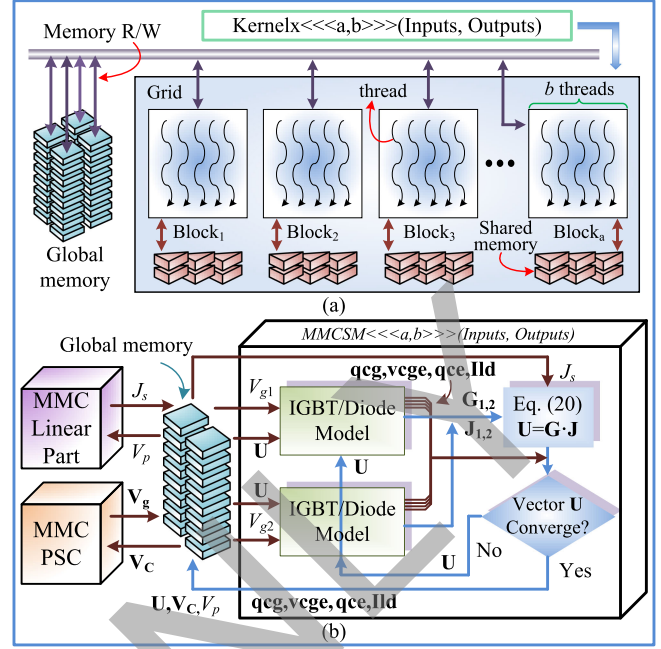


Fig. 7. GPU simulation architecture: (a) SIMT mode of kernel implementation and (b) nonlinear behavioral SM kernel.

taking the SM as an independent component instead of the entire MMC enables higher parallelism.

a) GPU computational architecture: The GPU kernel written in CUDA C programming language is a global function which has a similar coding principle to C/C++ programs run on the CPU. However, unlike its CPU counterpart which is implemented in a largely sequential manner, a kernel is enabled to have higher parallelism since the GPU has a much larger number of cores. In Fig. 7(a), the single-instruction-multiple-thread (SIMT) mode of GPU kernel implementation is given. The function *Kernelx* is computed in a grid of a blocks each having b threads, meaning a total number of $a \times b$ copies are launched. The inputs and outputs of a kernel are stored in global memory so that they can be accessed by other kernels, whilst inside a block, the shared memory can be declared. Within the kernel itself, these variables go to or come from all threads unless specified in the program. Thus, the GPU simulation is generally carried out in a way that circuit components having an identical attribute are written into a kernel, which, when invoked, conducts a massively parallel computation of all corresponding circuit components.

b) Nonlinear SM GPU kernel: The nonlinear behavioral IGBT/diode model-based SM has the largest quantity of all components that could be designed into a kernel. Therefore, when the GPU simulation is carried out, each SM corresponds to one thread, avoiding repeated calculation of the same function in CPU. The architecture of an HBSM is given in Fig. 7(b), and the FBSM has a similar form other than that the IGBT/diode model coded as a device function would be called by the SM kernel four times. Their outputs, i.e., the \mathbf{G} matrix and the vector \mathbf{J} , participate in circuit solution along with external components such as J_s . The Newton-Raphson iteration continues until the

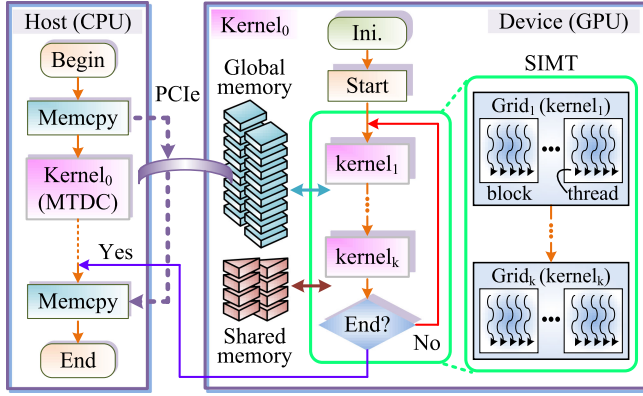


Fig. 8. Flowchart of the MTdc grid simulation on GPU using dynamic parallelism.

solution U converges, followed by updating history items and calculation of other outputs. The SM closely interacts with the MMC linear part and the controller, and all variables exchanged between kernels are stored in the global memory. The dimension of each signal is defined according to its actual number, e.g., in the MMC hierarchy, one arm corresponds to one current I_s , but when the SM calculation begins, the GPU kernel architecture determines that it can be accessed by all SMs. The CUDA dynamic parallelism [21] is featured since an outer kernel containing all three parts is designed to represent the dc grid.

D. Parallel Simulation Architecture

The CIGRÉ dc grid is written as one kernel $Kernel_0$, and with the GPU compute capability higher than sm_{35} , it is able to launch child kernels such as $kernel_1$ and $kernel_2$ denoting various components, as Fig. 8 shows. The host CPU does not conduct simulation as it works only at the beginning and end of the whole process when $Kernel_0$ is invoked and unidirectional data transfer occurs, which is completed immediately. The EMT solution is purely done on the GPU to avoid repeated data exchange with the CPU that would prolong the simulation. As can be seen, the child kernels are calculated in each time step until a predefined simulation time is reached. Therefore, the GPU computation time is virtually the simulation time. When invoked, each child kernel launches an individual compute grid containing a predesignated number of blocks and threads, and all components corresponding to the kernel are being computed concurrently.

The internal design algorithm of the MTdc kernel is given in Fig. 9, where the seven major child kernels are implemented sequentially as the outputs of one kernel can only be sent to its counterparts when it finishes computing. During initialization, all variables are stored in the global memory, and the simulation can start from the MMC controller which has three kernels at the bottom. After the IGBT gate pulses are generated, the four circuit kernels at the top are calculated. As for each kernel, it is implemented in the SIMT mode, with each thread representing one component. Thus, it is noted that slight differences between

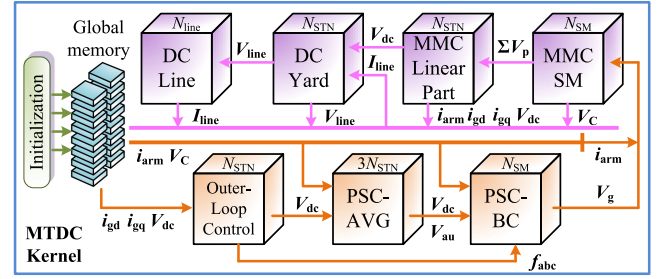


Fig. 9. MTdc grid GPU kernel design algorithm.

```
#pragma omp parallel for num_threads (NT)
for (int i=1; i< a×b; i++)
{
    Component (X[i], Y[i], Z[i], ...);
}
```

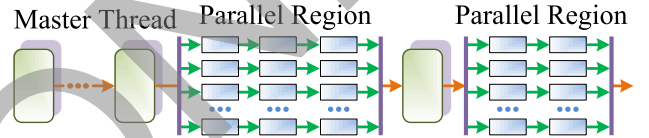


Fig. 10. OpenMP syntax and implementation flowchart.

components written as one kernel do not affect parallelism since they can be properly distinguished by the thread number.

CPU programs are also designed for comparison. The parallelism can be achieved if application programming interfaces (APIs) such as OpenMP is enabled on the multicore CPU, and the pseudocode for an arbitrary component type is shown in Fig. 10. The same variables in one component type are grouped as an array, so the C/C++ for loop can be used and subsequently the parallel OpenMP algorithm added outside the loop. As different component types have distinct sizes, there are various thread numbers in each parallel region, where sequential processing is quite often due to the limited core number of the CPU.

IV. GPU IMPLEMENTATION RESULTS AND VALIDATION

The GPU used in this work is the Nvidia Tesla V100 (Volta architecture) with 5120 CUDA cores and 16 GB HBM2 memory [27]. GPU implementation results at both device-level and system-level are demonstrated and validated by commercial off-line EMT solvers which run on a 64-bit Windows 10 operating system with 2.20 GHz 20-core Intel Xeon E5-2698 v4 CPU and 128 GB RAM.

A. Device-Level Switching Transients

The IGBT/diode modeling method is verified by the commercial device-level simulation tool SaberRD using its default Siemens IGBT module BSM300GA160D since it provides switch models that were experimentally validated, and the parameters are listed in the Appendix.

TABLE I
NBM-BASED MMC EXECUTION TIME BY VARIOUS PLATFORMS FOR 100-MS DURATION

MMC Level	Execution Time (s)			Speedup		
	SaberRD®	CPU ₁	GPU	SP ₁	SP ₂	SP ₃
5-L	709	56.2	159.1	12.6	4.5	0.35
7-L	1240	80.3	159.8	15.4	7.8	0.50
9-L	1720	98.4	163.7	17.5	10.5	0.60
11-L	—	121.2	163.3	—	—	0.74
21-L	—	260.3	206.0	—	—	1.26
33-L	—	368.9	238.4	—	—	1.55

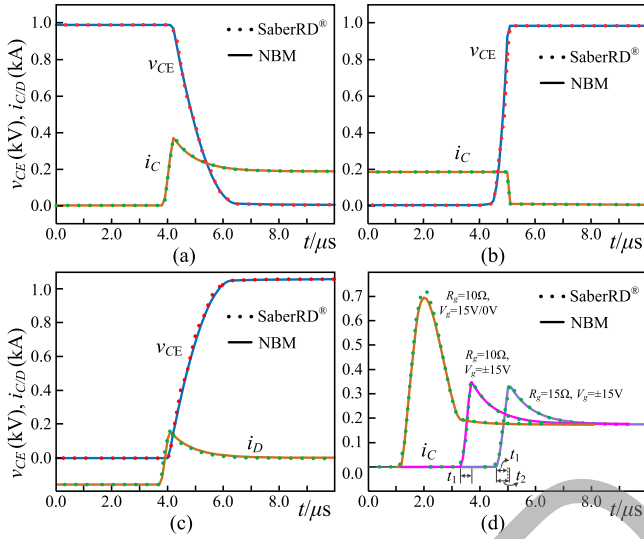


Fig. 11. Switching transients of behavioral IGBT/diode pair: (a) Turn-on, (b) turn-off, (c) diode reverse recovery, and (d) IGBT turn-on current under different gate conditions.

In **Table I**, the execution time of device-level simulation is compared by computing single-phase MMCs for a 100-ms duration with a 100-ns time step. It takes SaberRD up to 1700 s to compute a nine-level converter, and the results are no longer convergent once the voltage level reaches 11. Therefore, it is unfeasible for the device-level simulation package to conduct power system computation. In the meantime, the proposed IGBT/diode model and the decoupling method are also tested on the single-core CPU and the Nvidia Tesla V100 GPU. The proposed model enables the CPU to achieve a speedup SP_1 of almost 18 times in a 9-L MMC, and the speedup SP_2 by GPU is near 11. The GPU overtakes CPU when the MMC level reaches 21 since its speedup over CPU SP_3 is greater than 1.

Device-level results from a 9-L MMC with reduced dc bus voltage of 8 kV are given. With a dead-time $\Delta T = 5 \mu\text{s}$, a gate resistance of 10Ω and a voltage of $\pm 15 \text{ V}$, the switching transients are normal in **Fig. 11(a)–(c)**. Slight overshoot is observed in the IGBT turn-on current, and the diode reverse recovery process accounts for this phenomenon. **Fig. 11(d)** shows the impact of the gate-driving conditions on switching transients that is only available in device-level modeling. Adjusting the gate turn-off voltage to 0 V leads to a tremendous current overshoot, which means this driving condition is hazardous to the IGBT.

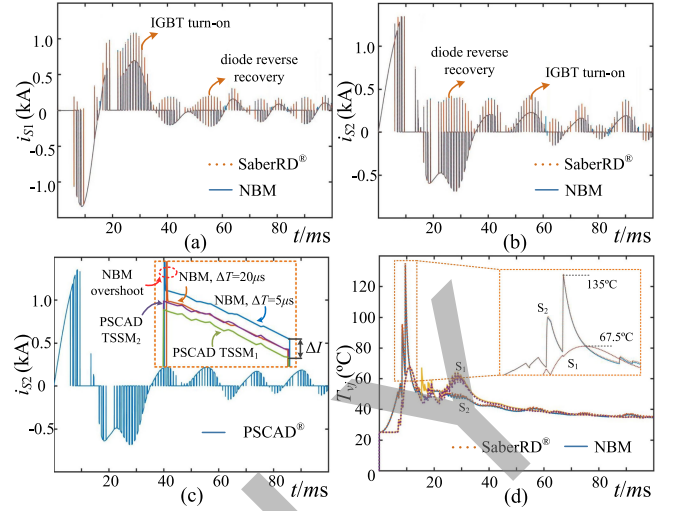


Fig. 12. Switching pattern and IGBT junction temperature: (a) Upper switch current, (b) lower switch current, (c) IGBT junction temperatures, and (d) switching pattern difference between device-level model and two-state switch model.

And when the gate resistance is set to 15Ω , the current rises more slowly as the time interval t_2 is slightly larger than t_1 . SaberRD simulation is also conducted, and a good agreement validates the proposed IGBT/diode nonlinear behavioral model and the designed MMC GPU kernel.

In **Fig. 12(a)–(c)**, the switching patterns between different tools are compared. The proposed NBM leads to exact static and dynamic current waveforms to SaberRD. In contrast, PSCAD/EMTdc is not able to give the actual current stress of an IGBT during operation, as the switching transients could not be observed. Moreover, the model also determines the simulation accuracy. It is shown by **Fig. 12(c)** that with default IGBT and diode model TSSM₁ and a typical time step of $20 \mu\text{s}$ in PSCAD/EMTdc, a current disparity $\Delta I = 6 \text{ A}$ out of 190 A is witnessed even under steady state. The result from PSCAD/EMTdc becomes closer to proposed NBM when an approximate ON-state resistance and voltage drop of the IGBT/diode pair is set to its switch model TSSM₂. The junction temperatures of the two complementary switches in a SM are given in **Fig. 12(d)**. A dramatic temperature surge is observed when the 9-L MMC starts to operate, and the curve decreases gradually along with the converter's entry into steady state. The correctness of these results is validated by SaberRD.

B. Wind Farm Integration Dynamics

The DCS1 subsystem is taken for illustration of 100 wind turbines' integration into the dc grid. **Fig. 13(a)** gives the power-wind speed characteristics of the DFIG. When the wind speed declines from 11 to 8 m/s in 1 s, as **Fig. 13(b)** indicates, the rotor mechanical velocity drops from the initial 188 rad/s to around 137 rad/s. Consequently, the rectifier-side currents are almost halved; nevertheless, the ac voltage remains virtually constant due to the proper control of MMC, as shown in **Fig. 13(c)** and **(d)**. In **Fig. 13(e)**, the output power of a single DFIG reduces from 1.96 MW to about 0.76 MW, which fits with the P - v

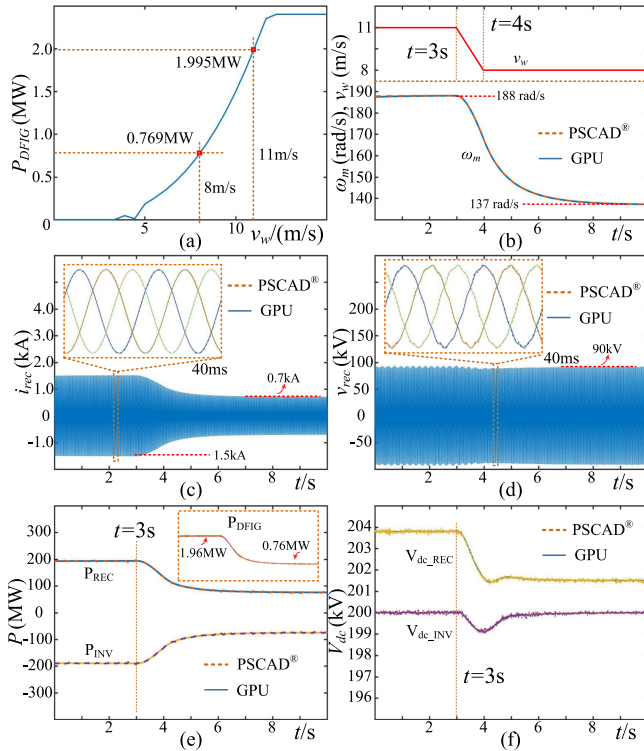


Fig. 13. OWF integration into DCS1: (a) DFIG P - v characteristics, (b) wind speed and rotor mechanical velocity, (c) rectifier ac currents, (d) rectifier ac voltage, (e) converter station dc yard power, and (f) terminal dc voltages.

characteristics. Therefore, the power at both stations gradually ramps down from 200 MW to about 76 MW. Fig. 13(f) demonstrates dc voltage fluctuation caused by the change in wind speed. The dc voltage at the inverter station has a momentary sag to 199 kV, but it recovers immediately. The rectifier-side dc voltage reduces as the power delivered between the two stations has a significant reduction. The above results are verified by PSCAD/EMTdc simulation.

C. MTdc System Tests

Fig. 14 gives the HBSM- and FBSM-MMC responses to dc line fault in DCS1. The pole-to-pole fault F_1 occurs to the center of the line at $t = 3$ s, and subsequently all IGBT gate signals are retrieved. It is shown in Fig. 14(a) that the dc current in FBSM case reduces to 0 after a few oscillations, while it eventually reaches over 10 kA with HBSM topology. Similarly, the FBSM achieves 0 kV on the dc line, but its counterpart is unable to block thoroughly since the freewheeling diodes operate as a rectifier. PSCAD/EMTdc simulation results are also given for validation. Minor differences are observed due to the adoption of different switch models, i.e., NBM and the TSSM. The fact that different ON-state resistances of the TSSM lead to distinct dc current and voltage waveforms indicates the importance of accurate switch models even in the system-level study.

The power flow of the entire grid under steady state is shown in Fig. 15 when OWF1–5 sends energy to the inland inverter stations. Cm-A1 in DCS1 receives virtually all 200 MW power

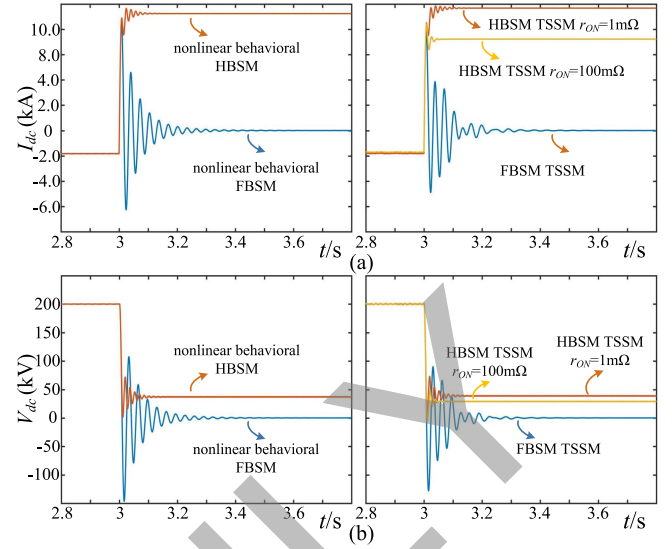


Fig. 14. Inverter-side HBSM- and FBSM-MMC response to dc fault (L: GPU simulation, R: PSCAD/EMTdc): (a) dc currents and (b) dc voltages.

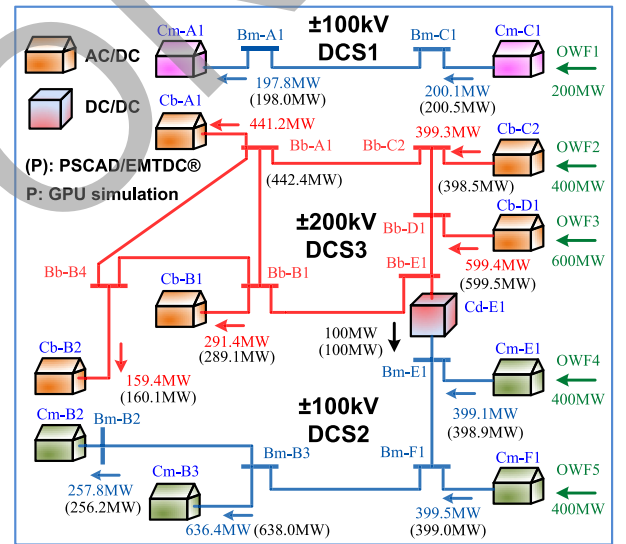


Fig. 15. Power flow in the CIGRÉ B4 dc grid.

from the OWF. In DCS2, the combined energy that Cm-B2 and Cm-B3 receive is around 100 MW more than that from OWF4 and OWF5 since Cd-E1 is ordered to deliver an additional 100 MW. As a consequence, Cb-A1, Cb-B1, and Cb-B2 have in total 892 MW while OWF2 and OWF3 send around 1 GW. The power distribution from PSCAD/EMTdc simulation shows virtually identical values.

Table II shows the time CPUs and the NVIDIA Tesla V100 GPU need to calculate the CIGRÉ B4 dc grid for 1-s duration with a time step of 200 ns. It can be seen that the single CPU is hardly able to simulate a practical dc system as it could take more than 1 million seconds. The situation is slightly improved by using multiple cores but they still require an extremely long period. In contrast, the V100 GPU can complete the simulation of 11 401-L MMCs in less than 1800 s, and there is no obvious difference in calculating HBSM-MMC or FBSM-MMC.

TABLE II
EXECUTION TIME OF CIGRÉ B4 DC GRID BY CPUS AND GPU FOR 1-S DURATION

MMC	Execution Time (s)						GPU Speedup			
	1 CPU core		20 CPU cores		V100 GPU		SP ₁ (over 1 CPU core)		SP ₂ (over 20 CPU cores)	
	Level	HBSM	FBSM	HBSM	FBSM	HBSM	FBSM	HBSM	FBSM	HBSM
51-L	2.29×10 ⁵	4.03×10 ⁵	2.64×10 ⁴	5.31×10 ⁴	901	908	254	444	29.3	58.5
101-L	4.42×10 ⁵	8.96×10 ⁵	5.81×10 ⁴	1.12×10 ⁵	957	957	462	936	60.7	117
201-L	9.47×10 ⁵	2.13×10 ⁶	1.14×10 ⁵	2.26×10 ⁵	1215	1218	779	1749	93.8	186
401-L	2.25×10 ⁶	4.51×10 ⁶	2.31×10 ⁵	4.59×10 ⁵	1728	1729	1302	2608	134	265

Consequently, the GPU gains a remarkable speedup over single CPU, i.e., 1302 and 2608 times when the MMC level is 401, and 134 and 265 over 20-core CPUs. As a further comparison, PSCAD/EMTdc was unable to compute the full-scale CIGRÉ dc grid even with much simpler IGBT and diode models.

Distinct speedups in Tables I and II reveal the advantage of GPU's massively parallel architecture as the system scales account for the discrepancy. The SM is the major factor that determines the simulation speed due to its complexity and quantity, and the SMs in the dc grid far outnumber that of MMCs in Table I. Although OpenMP is used for multicore CPU, the parallelism is much lower. Thus, the CPU needs far more sequential times to compute the CIGRÉ dc grid than a single MMC. In contrast, due to its sufficient parallel compute capability, the GPU simulation time rises much slower, and therefore a larger speedup can be gained.

V. CONCLUSION

The electromagnetic transient simulation of the CIGRÉ B4 dc grid integrated with OWFs employing nonlinear micromodeling of the MMC on the GPU was presented in this paper. High-order behavioral IGBT and diode models which have better accuracy were adopted in order to enable the simulation to reveal device-level information under various operation conditions correctly. The computational burden caused by the system-scale was first mitigated by fine-grained circuit partitioning of the MMC, which also improves numerical stability as the large number of small, identical circuit parts with nonlinear characteristics created subsequently are more convergent. The massively parallel architecture of the GPU was the key to further expedite the simulation. The partitioned SMs were designed into a GPU kernel, and by proceeding them with a corresponding number of computational threads conducting individual Newton–Raphson iterations separately instead of solving the entire MMC repeatedly, a tremendous speedup was attained. The simulation times required by both types of processors indicated that the GPU has the capability to conduct large-scale MTdc grid simulation using detailed nonlinear device-level models, which is hardly feasible by CPUs. Test results exhibited that the GPU has a huge potential in the future simulation of similar large-scale systems.

APPENDIX

The Siemens BSM300GA160D IGBT behavioral model parameters: $V_t = 6.3$ V, $v_{on} = 0.8$ V, $x = 0.974$, $y = 1.429$, $z = 0.369$, $a_1 = 0.022$, $b_1 = 0.004$, $r_{tail} = 1$ $\mu\Omega$, $C_{tail} = 10$ F,

$r_g = 5$ Ω , $i_{rat} = 0.05$; $L_d = 1e^{-11}$ H, $r_{on} = 0.01$ Ω , $r_{Ld} = 12.8$ $\mu\Omega$, $K_r = 9874$.

N-level MMC parameters: $C_{dc} = 100$ μ F, $C = N \cdot 100$ μ F, $L_{u,d} = 0.1$ p.u., converter transformer: Y-Y 270 kV/135 kV. DFIG parameters: $P_n = 2$ MW, $f_n = 50$ Hz, $V_s = 690$ V, $V_r = 2070$ V, $P_p = 2$. DC line: $R_{dis} = 12.1$ m Ω /km, $L_{dis} = 0.1$ mH/km, $C_{dis} = 0.3$ μ F/km.

REFERENCES

- [1] T. K. Vrana, S. Denetiere, Y. Yang, J. Jardini, D. Jovicic, and H. Saad, "The CIGRE B4 DC grid test system," *ELECTRA* issue 270, pp. 10–19, Oct. 2013.
- [2] Y. Wang, Z. Yuan, and J. Fu, "A novel strategy on smooth connection of an offline MMC station into MTDC systems," *IEEE Trans. Power Del.*, vol. 31, no. 2, pp. 568–574, Apr. 2016.
- [3] D. Tzelepis, G. Fusiek, A. Dyško, P. Niewczas, C. Booth, and X. Dong, "Novel fault location in MTDC grids with non-homogeneous transmission lines utilizing distributed current sensing technology," *IEEE Trans. Smart Grid*, to be published. doi:10.1109/TSG.2017.2764025.
- [4] A. Mokherdoran, D. V. Hertem, N. Silva, H. Leite, and A. Carvalho, "Multiport hybrid HVDC circuit breaker," *IEEE Trans. Ind. Electron.*, vol. 65, no. 1, pp. 309–320, Jan. 2018.
- [5] K. Rouzbehi, A. Miranian, J. I. Candela, A. Luna, and P. Rodriguez, "A generalized voltage droop strategy for control of multiterminal DC grids," *IEEE Trans. Indus. Appl.*, vol. 51, no. 1, pp. 607–618, Jan. 2015.
- [6] X. Chen, L. Wang, H. Sun, and Y. Chen, "Fuzzy logic based adaptive droop control in multiterminal HVDC for wind power integration," *IEEE Trans. Energy Convers.*, vol. 32, no. 3, pp. 1200–1208, Sep. 2017.
- [7] R. Dantas, J. Liang, C. E. Ugalde-Loo, A. Adamczyk, C. Barker, and R. Whitehouse, "Progressive fault isolation and grid restoration strategy for MTDC networks," *IEEE Trans. Power Del.*, vol. 33, no. 2, pp. 909–918, Apr. 2018.
- [8] 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.
- [9] N. Ahmed, L. Ångquist, S. Norrga, A. Antonopoulos, L. Harnefors, and H. P. Nee, "A computationally efficient continuous model for the modular multilevel converter," *IEEE J. Emerging Sel. Topics Power Electron.*, vol. 2, no. 4, pp. 1139–1148, Dec. 2014.
- [10] A. Beddard, M. Barnes, and R. Preece, "Comparison of detailed modeling techniques for MMC employed on VSC-HVDC schemes," *IEEE Trans. Power Del.*, vol. 30, no. 2, pp. 579–589, Apr. 2015.
- [11] A. M. Lopez, D. E. Quevedo, R. P. Aguilera, T. Geyer, and N. Oikonomou, "Limitations and accuracy of a continuous reduced-order model for modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 33, no. 7, pp. 6292–6303, Jul. 2018.
- [12] A. Christe and D. Dujic, "Virtual submodule concept for fast semi-numerical modular multilevel converter loss estimation," *IEEE Trans. Ind. Electron.*, vol. 64, no. 7, pp. 5286–5294, Jul. 2017.
- [13] L. Yang, Y. Li, Z. Li, P. Wang, S. Xu, and R. Gou, "A simplified analytical calculation model of average power loss for modular multilevel converter," *IEEE Trans. Ind. Electron.*, to be published. doi:10.1109/TIE.2017.2779417.
- [14] D. Tan, "Power electronics in 2025 and beyond: A focus on power electronics and systems technology," *IEEE Power Electron. Mag.*, vol. 4, no. 4, pp. 33–36, Dec. 2017.

- [15] A. R. Hefner, "A dynamic electro-thermal model for the IGBT," *IEEE Trans. Indus. Appl.*, vol. 30, no. 2, pp. 394–405, Mar. 1994.
- [16] M. Jin and M. Weiming, "Power converter EMI analysis including IGBT nonlinear switching transient model," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1577–1583, Oct. 2006.
- [17] S. Jahdi, O. Alatise, L. Ran, and P. Mawby, "Accurate analytical modeling for switching energy of PiN diodes reverse recovery," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1461–1470, Mar. 2015.
- [18] T. Kato, K. Inoue, T. Fukutani, and Y. Kanda, "Multirate analysis method for a power electronic system by circuit partitioning," *IEEE Trans. Power Electron.*, vol. 24, no. 12, pp. 2791–2802, Dec. 2009.
- [19] Z. Zhou and V. Dinavahi, "Fine-grained network decomposition for massively parallel electromagnetic transient simulation of large power systems," *IEEE Power Energy Technol. Syst. J.*, vol. 4, no. 3, pp. 51–64, Sep. 2017.
- [20] S. Yan, Z. Zhou, and V. Dinavahi, "Large-scale nonlinear device-level power electronic circuit simulation on massively parallel graphics processing architectures," *IEEE Trans. Power Electron.*, vol. 33, no. 6, pp. 4660–4678, Jun. 2018.
- [21] *CUDA C programming guide*, NVIDIA Corp., Santa Clara, CA, USA, Jun. 2017.
- [22] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff, *Analysis of Electric Machinery*. New York, NY, USA: IEEE Press, Jan. 1995.
- [23] M. Hagiwara and H. Akagi, "Control and experiment of pulsewidth-modulated modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 24, no. 7, pp. 1737–1746, Jul. 2009.
- [24] H. Abu-Rub, M. Malinowski, and Kamal Al-Haddad, *Power Electronics for Renewable Energy Systems, Transportation and Industrial Applications*. NJ, USA: Wiley-IEEE Press, 2014.
- [25] M. Zhang, A. Courtay, and Z. Yang, "An improved behavioral IGBT model and its characterization tool," *Proc. IEEE Hong Kong Electron Devices Meeting*, 2000, pp. 142–145.
- [26] A. Courtay, "MAST power diode and thyristor models including automatic parameter extraction," *SABER User Group Meeting.*, Brighton, U.K., pp. 1–10, Sep. 1995.
- [27] *Whitepaper NVIDIA Tesla V100 GPU Architecture*, NVIDIA Corp., NVIDIA Corp., Santa Clara, CA, USA, Aug. 2017.



field-programmable gate arrays.

Ning Lin (S'17) received the B.Sc. and M.Sc. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2008 and 2011, respectively. He is currently working toward the Ph.D. degree in electrical and computer engineering with the University of Alberta, Edmonton, AB, Canada.

He was a Research Engineer on FACTS and HVdc control and protection in China. His research interests include real-time simulation, parallel simulation of power electronics, and



Venkata Dinavahi (S'94–M'00–SM'08) received the B.Eng. degree in electrical engineering from the Visveswaraya National Institute of Technology (VNIT), Nagpur, India, the M.Tech degree in electrical engineering from the Indian Institute of Technology (IIT) Kanpur, India, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Ontario, Canada, in 1993, 1996, and 2000, respectively.

He is currently a Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems and power electronic systems, electromagnetic transients, device-level modeling, large-scale systems, and parallel and distributed computing.