

# A Fast and Stable Method for Modeling Generalized Nonlinearities in Power Electronic Circuit Simulation and Its Real-Time Implementation

Zhen Huang , *Student Member, IEEE*, and Venkata Dinavahi , *Senior Member, IEEE*

**Abstract**—Nonlinearities have been the major obstructions that limit the computational efficiency in power electronic circuit simulation for a long time. Yet there is no standard way for dealing with them. This paper presents a new method that makes the handling of nonlinearities fast and stable. In the proposed method, nonlinearities are transformed into a uniform representation—a constant resistor in parallel with a companion current source, thus making the system admittance matrix constant for fixed time-step simulation. To solve for the corresponding companion current source, nonlinearities are treated as either current or voltage sources and a diagonal time-varying matrix equation is developed. Three methods are proposed for solving the matrix equation—precomputed inversion or factorization, modified Gaussian elimination, and updating inverse using the Sherman–Morrison formula—that can fit different system sizes and applications. The proposed method is validated by two common power electronic converter topologies, both in offline and real-time simulation. Offline tests show that the proposed method achieved the same accuracy with the mature simulation software while being more than ten times faster. The same test cases are also implemented into field programmable gate arrays based real-time simulation experiments for verification.

**Index Terms**—Admittance matrix, electromagnetic transient analysis, field programmable gate arrays (FPGA), nonlinearities, real-time systems.

## I. INTRODUCTION

MODERN power electronic systems need more powerful simulation tools to meet the demands of speed and accuracy as the system complexity increases accordingly. Some systems, such as the modular multilevel converter (MMC), contain hundreds of components, including both linear (capacitors and inductors) and nonlinear (semiconductor switches) components in one circuit instance. The large number of switching elements in the MMC introduces a challenge for modeling the converter on electromagnetic transient simulation programs [1]. The linear components can be dealt with properly

using numerical integration methods, such as backward Euler or the trapezoidal method, whereas the nonlinear components still pose the bottleneck of simulation performance.

Nonlinearities bring uncertainties into the system admittance matrix for nodal analysis and should be taken care of differently according to their characteristics [2]. Basically, there are two ways to deal with nonlinearities: iterative and noniterative. The iterative methods combine the linear network equations with the nonlinear characteristic equations and try to find a solution that satisfies both of them. Some iterative algorithms, such as the Newton–Raphson method, are utilized to search for the solution. The iterative solution (if it exists) is always theoretically accurate (i.e., no approximations are made). One example of an iterative model is the zinc oxide arrester model in the EMTP [3] and the online monitoring program in [4], which uses an analytic equation to model its behavior. Other examples can be found in the works of [5]–[8]. However, there are mainly two drawbacks to iterative methods. First, there is no theoretical guarantee that the iteration solution can be found; in some cases, the iteration methods do not converge. Second, the iterative search is usually time consuming for a required accuracy because the convergence speed varies case by case. Thus, iterative approaches are typically employed in high accuracy and nontime critical applications, usually offline simulation for a limited system size. Some simulation tools, such as SaberRD, Pspice, and PAN academic circuit simulator (described in [9]), belong to this category.

Noniterative methods usually sacrifice some accuracy to trade for simulation speed so that they can be utilized in time-critical applications, such as real-time simulation or to handle larger systems. There are numerous effective schemes to simplify nonlinearities. Many of them are designed based on the specific nonlinearity's characteristics, and therefore cannot be promoted as general methods. Among these approaches, two of the most commonly used are: current-source representation with time lag  $\Delta t$  and piecewise linear representations [3]. The first approach is easy to understand and implement. At simulation time  $t$ , all variables are known at  $t - \Delta t$  (here a fixed time-step method is assumed); thus, they can be used to calculate the current flow of nonlinear element at time  $t - \Delta t$  and inject it as a current source at time  $t$ . Some very complicated or large-inertia nonlinear elements are implemented using this way, such as electric machines in PSCAD/EMTDC [10] and the work in [11]. However, under this principle, any sudden change in voltage causes a current

Manuscript received January 10, 2018; revised April 27, 2018; accepted June 22, 2018. Date of publication June 28, 2018; date of current version February 20, 2019. This work was supported by the Natural Science and Engineering Research Council of Canada. The work of Z. Huang was supported by the China Scholarship Council. Recommended for publication by Associate Editor F. H. Khan. (*Corresponding author: Zhen Huang.*)

The authors are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2R3, Canada (e-mail: zh2@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/TPEL.2018.2851570

response only in the next time step. Thus, for the previous time step, the machine looks like an open circuit and spurious spikes may appear in the machine terminal voltage [12]. In order to alleviate this phenomenon, some modifications have to be made. In [12], a resistance is placed in parallel with the machine to reduce the spurious spikes and an additional compensated current source is added to offset the current introduced by this resistance. The current source representation with time lag  $\Delta t$  runs fast in simulation and is accurate provided that the time step is small enough. This condition, however, is not difficult to meet as the digital processor performance has increased dramatically over the years.

The piecewise linear approach divides the nonlinear element characteristics into several working sections and models it as a linear component with different parameters to fit its characteristic in each section. Typical example of this method is the simple switch model in PSCAD/EMTDC. During ON state, it acts as a very small resistance while in OFF state a very large resistance [10]. Other piecewise linear model applications can be found in the works presented in [13]–[18]. This approach can achieve a reasonable accuracy while retaining an elegant mathematic form. However, the varying parameters will alter the system admittance matrix and each time it is changed, the triangularization or inversion process has to be reconducted, thus lowering the simulation speed. If there are lots of nonlinear elements in a system, such as the MMC, the execution rate can be extremely low; therefore, making this approach impractical. There are some efforts to make the switch model equal for both ON and OFF states [19]–[20] so as to make the admittance matrix constant. However, such methods either induce inaccurate power losses or require complicated revising calculations, and cannot be promoted to model other nonlinearities. Pejović and Maksimović [21] present a method to simulate the piecewise linear elements using constant admittance matrix and maintain speed advantage over general purpose simulation tools; however, the equivalent resistor's value and the computation of companion current source of piecewise linear elements are not addressed fully.

These two approaches have covered the most scope of non-iterative nonlinear element modeling, yet they cannot replace one another. For components like electric machines, it is very difficult to find their equivalent linear circuits because their characteristics change continually with rotor position, angular velocity, magnetic flux saturation, etc. On the other hand, components like switches are not suitable for current source representations with time lag  $\Delta t$  because during ON state, they behave more like a voltage source (i.e., zero terminal voltage with almost arbitrary current). Due to the limitation of these two modeling methods, they have to coexist in one simulation; however, the different mathematic representations make the solution process complex and lower the execution efficiency.

Some other methods decouple the nonlinear components manually so that they can be dealt independently [22]–[25]. These include the latency insertion method and transmission line modeling method. Such methods introduce artificial latency into a system (by inserting small inductive or capacitive elements) so as to permit a complete (approximate) decoupling

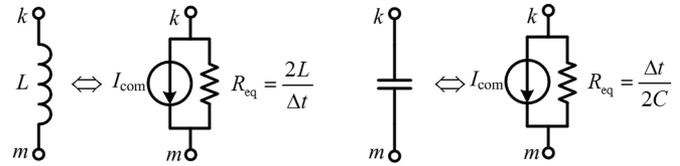


Fig. 1. Norton equivalent circuit of lumped inductance and capacitance.

of the system equations [22]. The side effect is that the inserted inductive or capacitive elements may distort the original component's behavior when its time constant is very small. Therefore, sometimes it is very difficult to choose the appropriate latency or transmission line parameters.

In this paper, a generalized method for modeling nonlinear elements in power electronic circuit simulation is presented. This method can deal with any form of nonlinear characteristic using a uniform mathematic representation, thus making the admittance matrix constant. By properly selecting the equivalent resistor, the time-varying part of the matrix to solve nonlinear elements is restricted only on diagonals. Three methods to solve the diagonal time-varying matrix equations are presented correspondingly. These properties can significantly reduce the execution burden of the simulation program. In addition, the system is decoupled naturally and only when there are real inductances or capacitances; thus, high accuracy is retained. Besides, the method does not have instability risk because it is inherently stable. Offline and real-time simulation results of two typical power electronic circuits—MMC and neutral point clamped (NPC) topologies—are provided and compared with the existing simulation softwares both at system level and device level. This paper is organized as follows. Sections II and III introduce the uniform representation of nonlinear elements and the classification of nonlinear behaviors. Section IV elaborates the solution process of nonlinear elements' companion current source. Sections V and VI present the offline and real-time simulation validation results and Section VII concludes this paper.

## II. UNIFORM REPRESENTATION OF NONLINEAR ELEMENTS

Most of the circuit simulation programs choose node voltages as the unknown variables and use nodal analysis to develop a set of linear equations by applying Kirchhoff's current law (KCL) at every circuit node. In nodal analysis, it is preferable to use the Norton equivalent circuit (a current source in parallel with a resistive branch) to represent the circuit element. Fig. 1 shows the Norton equivalent representation of lumped linear inductance and capacitance.

The equivalent resistor  $R_{eq}$  is obtained using the trapezoidal rule of integration and it is a constant for a fixed time-step  $\Delta t$ . The companion current source  $I_{com}$  is calculated from the past history (the previous time-step variables) [2]. However, when this concept is applied to nonlinear elements, usually both  $R_{eq}$  and  $I_{com}$  are time varying; thus, making the analytical solution process complex and the computational complexity of

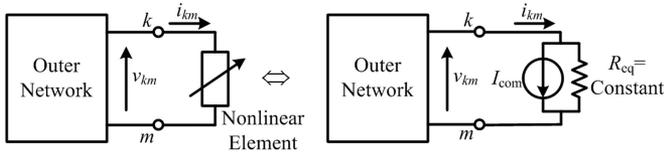


Fig. 2. Norton equivalent circuit of nonlinear element.

solving a time-varying matrix equation is  $O(N^3/3)$ . Actually, the potential of  $I_{com}$  has not been exploited fully yet.  $I_{com}$  can be calculated not only from history variables but also from current time-step variables. Assume that there is a nonlinear element connecting an external network at node  $k$  and  $m$ , as shown in Fig. 2. No matter what kind of characteristics this nonlinear element possesses, ultimately there exists a solution for terminal voltage  $v_{km}(t)$  and branch current  $i_{km}(t)$  at simulation time  $t$ . From the viewpoint of the external network, as long as (1) is satisfied, it cannot distinguish whether the component between nodes  $k$  and  $m$  is a nonlinear element or a constant resistor  $R_{eq}$  in parallel with a current source  $I_{com}$

$$I_{com}(t) = i_{km}(t) - \frac{v_{km}(t)}{R_{eq}}. \quad (1)$$

As can be seen from Fig. 2, the equivalent resistor for the nonlinear element is a constant, just like the linear element in a fixed time-step simulation. Thus, the system admittance matrix becomes constant since only the equivalent resistor will appear in the admittance matrix. As a result, the computational complexity for solving node voltages reduces to  $O(N^2)$ .

The only problem of applying this representation for nonlinear elements is how to find this  $I_{com}(t)$  that satisfies (1) since both  $v_{km}(t)$  and  $i_{km}(t)$  are unknown. Actually, (1) is difficult to solve because not only the relationship between  $v_{km}(t)$  and  $i_{km}(t)$  is nonlinear but also that the external network may contain multiple nonlinear components. The theoretic accurate solution cannot be found without using iterative methods. In order to save execution time, approximations have to be made like all other noniterative methods.

### III. CLASSIFICATION OF NONLINEAR BEHAVIOR

Before introducing the appropriate approximations that should be made to match this uniform representation, the nonlinear behaviors are classified first. In the authors' opinion, nonlinear behavior can be roughly classified into two categories: current-source-type behavior and voltage-source-type behavior. Some nonlinear elements always behave as one type, but others may transfer types according to their working condition. The classification is done mainly by empirical conventions and can be programmed into the model as an attribute for reuse. Table I lists some conventions of nonlinearities.

#### A. Current-Source-Type Behavior

Typical current-source-type behaviors include the output characteristics of PV panel, an electric machine under load

TABLE I  
SOME CONVENTIONS OF NONLINEARITIES

Voltage-source-type	Current-source-type
Batteries, Generators Switches in turn-ON transients and steady ON-state	PV panels, Motors, Switches in turn-OFF transients and steady OFF-state

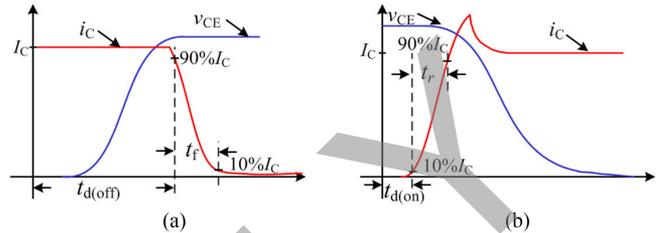


Fig. 3. Switching transients of semiconductor switches. (a) Turn-OFF transient. (b) Turn-ON transient.

mode, a semiconductor switch in turn-OFF transient and steady OFF state, etc. What they have in common is that they can be viewed as a current source in the circuit, and more importantly, their current can be predicted precisely based on the history (for example, the previous time step) information. Take the semiconductor switch as an example, the turn-OFF transient characteristics can be obtained experimentally and recorded to develop a turn-OFF analytical function, as shown in Fig. 3(a). Some of the characteristic parameters, such as turn-OFF delay time  $t_{d(off)}$  and fall time  $t_f$ , can be extracted from the experimental data. Finally, a per-unit turn-OFF function can be obtained and used to generate the actual waveforms by scaling with the current amplitude in simulation. The key point of this method is to determine the current value at the next time step by a predefined trajectory and this trajectory comes from many experimental measurements to achieve a high precision. This approach has a very good curve-fitting performance and offers reasonable accuracy for current prediction in turn-OFF transient [11], [26]. As for the steady OFF state, the switching device can be viewed as a current source whose current is zero. Other elements, such as PV panel and electric machine, do not have a predefined trajectory. Their current can be predicted simply because of system inertia. Provided that the time step is small enough, their current can be updated using previous time-step information and assumed to be consistent over one time step.

#### B. Voltage-Source-Type Behavior

Typical voltage-source-type behaviors include the output characteristics of most kinds of batteries, an electric machine under generator mode, semiconductor switches in turn-ON transient and steady ON state, etc. What they have in common is that they can be viewed as a voltage source in the circuit, and more importantly, their voltage can be predicted precisely based on the history information. For example, the voltage trajectory of a semiconductor switch in turn-ON transient can be tested and recorded to develop a per-unit function. When scaled with the

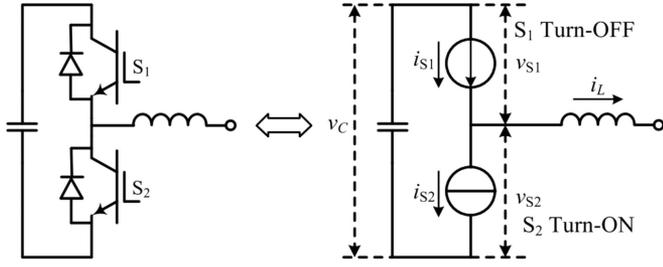


Fig. 4. Switching transient analysis of two complementary semiconductor switches.

voltage amplitude in simulation, it can predict the voltage value at the next time step. Other elements' voltages, such as batteries and electric machines, can be predicted because of their system inertia.

One may ask, since the voltage and current trajectories of semiconductor switches can be recorded simultaneously in experiments, why under turn-OFF transient should it be treated as a current source and under turn-ON transient as a voltage source? That is because under turn-OFF transient, the conducting current will decrease from some value to zero, which means that the initial and final values of current are known during this period. Then, the initial current value can be used as the scaling factor of the per-unit function. While in the same period, the voltage across the switch will increase from almost zero to some value that the program does not know. The voltage per-unit function is useless without an appropriate scaling factor. Actually, the voltage waveform of a semiconductor switch under turn-OFF transient is usually determined by its complementary device's voltage turn-ON trajectory. This can be illustrated in Fig. 4. If  $S_1$  is under turn-OFF transient, then  $S_2$  or its antiparallel diode will be under turn-ON transient.  $S_1$  will act as a current source and  $S_2$  a voltage source. The current of  $S_2$  is determined by the difference of  $i_L$  and  $i_{S1}$  while the voltage of  $S_1$  is determined by the difference of  $v_C$  and  $v_{S2}$ .

#### IV. SOLUTION PROCESS OF NONLINEAR ELEMENTS' COMPANION CURRENT SOURCES

Assume that all the elements in the system to be simulated are represented by their Norton representations, and the equivalent resistor is constant for linear and nonlinear components. Then, a set of node voltage equations can be formed by applying KCL as the matrix equality shown in the following equation:

$$\mathbf{G}\mathbf{V} = \mathbf{I}' \quad (2)$$

where  $\mathbf{G}$  is a constant nodal conductance matrix,  $\mathbf{V}$  is the node voltage vector, and  $\mathbf{I}'$  is the companion current source vector. Note that the members of  $\mathbf{I}'$  are not independent companion current sources but their linear combinations. For example, the equivalent circuit of the half-bridge topology in Fig. 4 is shown in Fig. 5 and the KCL equation at node ② can be expressed as

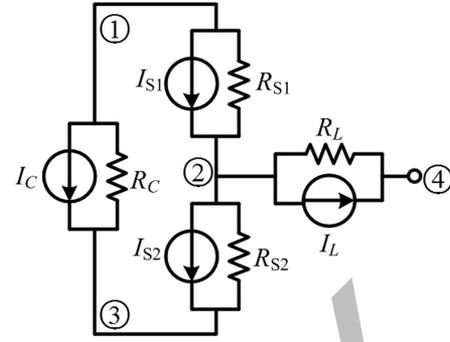


Fig. 5. Equivalent circuit of the half-bridge topology.

follows:

$$-\frac{1}{R_{S1}}v_1 + \left(\frac{1}{R_{S1}} + \frac{1}{R_{S2}} + \frac{1}{R_L}\right)v_2 - \frac{1}{R_{S2}}v_3 - \frac{1}{R_L}v_4 = I_{S1} - I_{S2} - I_L. \quad (3)$$

The right-hand side of (3) can be further transformed as a constant vector multiplying the independent companion current source vector  $\mathbf{I}$  shown in the following equation:

$$I_{S1} - I_{S2} - I_L = [1 \quad -1 \quad 0 \quad -1] \begin{bmatrix} I_{S1} \\ I_{S2} \\ I_C \\ I_L \end{bmatrix}. \quad (4)$$

Assume that there are  $N_1$  nodes and  $N_2$  independent companion current sources in the system, then the matrix equality in (2) can be recomposed as following:

$$\mathbf{G}_{N_1 \times N_1} \mathbf{V}_{N_1 \times 1} = \mathbf{I}'_{N_1 \times 1} = \mathbf{T}_{N_1 \times N_2} \mathbf{I}_{N_2 \times 1} \quad (5)$$

where each row of  $\mathbf{T}$  indicates the current source configuration at one node.  $\mathbf{T}$  is also constant and its elements are either 0 or 1 or  $-1$ . Based on (5), the node voltage vector can be expressed as

$$\mathbf{V}_{N_1 \times 1} = \mathbf{G}_{N_1 \times N_1}^{-1} \mathbf{T}_{N_1 \times N_2} \mathbf{I}_{N_2 \times 1} = \mathbf{A}_{N_1 \times N_2} \mathbf{I}_{N_2 \times 1} \quad (6)$$

where  $\mathbf{A}_{N_1 \times N_2} = \mathbf{G}_{N_1 \times N_1}^{-1} \mathbf{T}_{N_1 \times N_2}$ . Note that  $\mathbf{A}_{N_1 \times N_2}$  is constant as well since both  $\mathbf{G}_{N_1 \times N_1}$  and  $\mathbf{T}_{N_1 \times N_2}$  are constant.

Equation (6) reveals an important piece of information: given that all the components in system are represented by constant resistors in parallel with current sources, the node voltage can be expressed as linear combinations of these current sources, and more importantly, the coefficients of these current sources are constant. This is also true for any voltage difference between two nodes. For example, the voltage difference between nodes  $k$  and  $m$  at simulation time  $t$  can be expressed as

$$v_{km}(t) = \sum_{j=1}^{N_2} (A_{k,j} - A_{m,j}) I_j(t). \quad (7)$$

Furthermore, the branch current between nodes  $k$  and  $m$  is given as

$$\begin{aligned} i_{km}(t) &= \frac{v_{km}(t)}{R_{\text{eq}}} + I_l(t) \\ &= \frac{\sum_{j=1}^{N_2} (A_{k,j} - A_{m,j}) I_j(t)}{R_{\text{eq}}} + I_l(t) \end{aligned} \quad (8)$$

where  $l$  is the index number of  $I_{\text{com}}$  between nodes  $k$  and  $m$  in the system.

As discussed in Section III, the nonlinear element will behave like a current or voltage source according to its nature or working condition and its current or voltage value at the next time step can be predicted with reasonable precision. Then, either  $i_{km}(t)$  or  $v_{km}(t)$  in (1) can be replaced with the predicted value. This is the approximation that is made to simplify the solution process of nonlinearities'  $I_{\text{com}}(t)$ .

Assume that there are  $N_3$  nonlinear elements or pure voltage/current sources labeled as 1 to  $N_3$  (here pure voltage/current sources and nonlinear elements are categorized into one group since they have no differences from the point view of the simulation program) and  $N_4$  linear elements labeled as  $N_3 + 1$  to  $N_2$  in system ( $N_4 = N_2 - N_3$ ). If the simulation program has finished the calculations at time  $t - \Delta t$  and is going forward to time  $t$ , the companion current source of  $N_4$  linear elements can be computed first based on information at time  $t - \Delta t$ . The companion current source of the other  $N_3$  elements should be solved based on (1). For each of these  $N_3$  elements, if  $i_{km}(t)$  is known, then replacing  $v_{km}(t)$  with (7) to have

$$I_l(t) = i_{km}(t) - \frac{\sum_{j=1}^{N_2} (A_{k,j} - A_{m,j}) I_j(t)}{R_{\text{eq}}} \quad (9)$$

which can be rewritten as

$$\begin{aligned} \frac{1}{R_{\text{eq}}} \sum_{j=1}^{N_3} (A_{k,j} - A_{m,j}) I_j(t) + I_l(t) \\ = i_{km}(t) - \frac{1}{R_{\text{eq}}} \sum_{j=N_3+1}^{N_2} (A_{k,j} - A_{m,j}) I_j(t). \end{aligned} \quad (10)$$

If  $v_{km}(t)$  is known, then replacing  $i_{km}(t)$  with (8) results in

$$I_l(t) = \frac{\sum_{j=1}^{N_2} (A_{k,j} - A_{m,j}) I_j(t)}{R_{\text{eq}}} + I_l(t) + \frac{v_{km}(t)}{R_{\text{eq}}} \quad (11)$$

which can in turn be rewritten as

$$\begin{aligned} \sum_{j=1}^{N_3} (A_{k,j} - A_{m,j}) I_j(t) \\ = v_{km}(t) - \sum_{j=N_3+1}^{N_2} (A_{k,j} - A_{m,j}) I_j(t). \end{aligned} \quad (12)$$

In Section II, the equivalent resistor  $R_{\text{eq}}$  of nonlinear element is only assumed to be constant but is not assigned any specific value. In this paper, all the  $R_{\text{eq}}$  of nonlinearities are chosen to be 1 so as to make (10) and (12) to have similar form. As can be seen

that if  $R_{\text{eq}}=1$ , then apart from the additional  $I_l(t)$ , the left-hand side of (10) and (12) is exactly the same, and their right-hand sides are all known items. Therefore,  $N_3$  linear equations are developed and can be written as

$$\mathbf{M}_{N_3 \times N_3} \mathbf{I}_{N_3 \times 1} = \mathbf{b}_{N_3 \times 1}. \quad (13)$$

Each row of  $\mathbf{M}_{N_3 \times N_3}$  represents a current or voltage equation of one nonlinear element or pure voltage/current source. Coefficients of  $\mathbf{M}_{N_3 \times N_3}$  can be computed by doing row subtraction of matrix  $\mathbf{A}_{N_1 \times N_2}$ . The only thing that one has to be aware of is that an additional 1 has to be added to the diagonals of  $\mathbf{M}_{N_3 \times N_3}$  if that row represents a current equation like (10). This means that the off-diagonal elements of matrix  $\mathbf{M}_{N_3 \times N_3}$  are always constant, the only time-varying part is its diagonal part.

As mentioned in Section III, some nonlinear elements always behave like either current or voltage source and will not transfer types during simulation. These include PV panel, batteries, some electric machines, pure voltage/current sources, etc. Other nonlinear elements, such as semiconductor switches, will consistently transfer types between current and voltage sources. Therefore,  $\mathbf{M}_{N_3 \times N_3}$  can be partitioned into blocks and (13) can be rewritten as

$$\begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{I}_1 \\ \mathbf{I}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \quad (14)$$

where the upper equations represent the elements that will change types during simulation and the lower equations represent those that do not. This means  $\mathbf{M}_{12}$ ,  $\mathbf{M}_{21}$ , and  $\mathbf{M}_{22}$  are constant, and only the diagonal of  $\mathbf{M}_{11}$  is time varying. Using lower part of (14),  $\mathbf{I}_2$  can be expressed as

$$\mathbf{I}_2 = -\mathbf{M}_{22}^{-1} \mathbf{M}_{21} \mathbf{I}_1 + \mathbf{M}_{22}^{-1} \mathbf{b}_2. \quad (15)$$

Replacing  $\mathbf{I}_2$  into the upper part of (14) using (15) yields the following system equations:

$$(\mathbf{M}_{11} - \mathbf{M}_{12} \mathbf{M}_{22}^{-1} \mathbf{M}_{21}) \mathbf{I}_1 = \mathbf{b}_1 - \mathbf{M}_{12} \mathbf{M}_{22}^{-1} \mathbf{b}_2. \quad (16)$$

Equation (16) is the core difficulty of solving the whole system states since this is the only process involving time-varying matrix. For the sake of convenience, in the remaining part of this paper, (16) will be denoted as

$$\mathbf{M} \mathbf{I} = \mathbf{b} \quad (17)$$

where  $\mathbf{M} = \mathbf{M}_{11} - \mathbf{M}_{12} \mathbf{M}_{22}^{-1} \mathbf{M}_{21}$  has time-varying diagonal and constant off-diagonal elements.  $\mathbf{b} = \mathbf{b}_1 - \mathbf{M}_{12} \mathbf{M}_{22}^{-1} \mathbf{b}_2$  is known based on history information.

The following sections provide three feasible methods for solving (17).

#### A. Solution by Precomputed Inversion or Factorization

When the dimension of  $\mathbf{M}$  is small, it is very convenient to compute all the possible cases of  $\mathbf{M}$  and store its inversion or factorization in memory. The corresponding computational complexity is constant. It has to be pointed out that this method is not necessarily only restricted to systems with very few elements. Some topologies, such as MMC, are also suitable

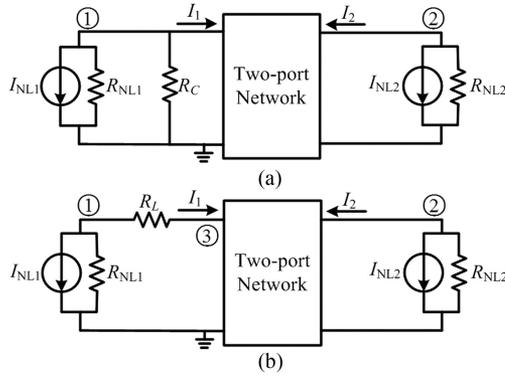


Fig. 6. Two configurations of decoupling elements.

for this method. To demonstrate this, an important property is indicated and proved here—when a very small resistor lies in parallel or a very large resistor lies in series with the connecting route of two nonlinear elements, then the voltages and currents of these two nonlinear elements are almost decoupled. Since the equivalent resistor of an inductor ( $2L/\Delta t$ ) is usually very large and the equivalent resistor of a capacitor ( $\Delta t/2C$ ) is usually very small, this property is useful in application.

Assume that there is a very small resistor  $R_C$  that lies in parallel with the route between two nonlinear elements, as shown in Fig. 6(a). The remaining part of the route is assembled in a two-port network and described by a  $Y$  matrix shown as follows:

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}. \quad (18)$$

Then, the following KCL equations can be developed:

$$\begin{bmatrix} \frac{1}{R_{NL1}} + \frac{1}{R_C} + y_{11} & y_{12} \\ y_{21} & \frac{1}{R_{NL2}} + y_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -I_{NL1} \\ -I_{NL2} \end{bmatrix}. \quad (19)$$

Solving (19),  $v_1$  and  $v_2$  can be expressed as

$$\begin{aligned} v_1 &= \frac{1}{\Delta} \left[ -\left( \frac{1}{R_{NL2}} + y_{22} \right) I_{NL1} + y_{12} I_{NL2} \right] \\ v_2 &= \frac{1}{\Delta} \left[ y_{21} I_{NL1} - \left( \frac{1}{R_{NL1}} + \frac{1}{R_C} + y_{11} \right) I_{NL2} \right] \end{aligned} \quad (20)$$

where  $\Delta$  is the determinant of the matrix in (19)

$$\Delta = \left( \frac{1}{R_{NL1}} + \frac{1}{R_C} + y_{11} \right) \left( \frac{1}{R_{NL2}} + y_{22} \right) - y_{12} y_{21}. \quad (21)$$

Because  $R_C$  is very small and  $\Delta$  is a very large value. Then, the cross-coupling coefficients  $y_{12}/\Delta$  and  $y_{21}/\Delta$  in (20) are almost zero, which means  $I_{NL1}$  has negligible effect on  $v_2$  and  $I_{NL2}$  has negligible effect on  $v_1$ . Thus,  $v_1$  and  $v_2$  are almost decoupled.

Similarly, when there is a very large resistor  $R_L$  that lies in series with the route between two nonlinear elements, as shown

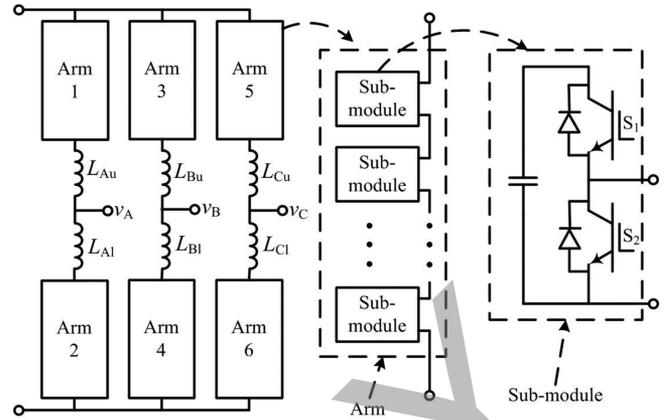


Fig. 7. Structure of an MMC topology.

in Fig. 6(b), following equations can be developed:

$$\begin{bmatrix} \frac{1}{R_{NL1}} + \frac{1}{R_L} & 0 & -\frac{1}{R_L} \\ 0 & \frac{1}{R_{NL2}} + y_{22} & y_{21} \\ -\frac{1}{R_L} & y_{12} & \frac{1}{R_L} + y_{11} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -I_{NL1} \\ -I_{NL2} \\ 0 \end{bmatrix}. \quad (22)$$

The cross-coupling coefficient between  $v_1$  and  $I_{NL2}$  is  $\frac{y_{12}}{\Delta} \frac{1}{R_L}$ , and the cross-coupling coefficient between  $v_2$  and  $I_{NL1}$  is  $\frac{y_{21}}{\Delta} \frac{1}{R_L}$ , where  $\Delta$  is the determinant of the matrix in (22). Because  $R_L$  is very large,  $\Delta$  is a normal value while  $\frac{1}{R_L}$  is almost zero. Thus,  $v_1$  and  $v_2$  are almost decoupled. It can also be concluded that the larger the values of  $L$  and  $C$  and the smaller the time step, the higher the decoupling degree will be. This property does indeed coincide with the objective laws in a real system—large inductors and capacitors are usually used to add inertia into the system and smaller time steps will offer better precision. Actually, there are already some efforts in literature using the inductive and capacitive parts to decouple the circuits and they often explained it in physics. This section provides a way to explain in mathematics why inductive and capacitive parts can be used to decouple the circuits and why inductive element should lie in series and capacitive element should lie in parallel with the decoupled subcircuits.

The MMC topology has a very elegant structure, as shown in Fig. 7. A three-phase MMC has six arms, each arm has several submodules (SMs) and one arm inductor, and each SM consists of one capacitor and two switches in a half-bridge topology. The arm inductors are used to limit the circulating current within the converter. However, they also provide great facility in simulation. As can be seen in Fig. 7, the connecting route of any two SMs has two series-connected arm inductors, which means these SMs are almost decoupled from each other and can be dealt separately. Since all the SMs in the MMC share the same  $2 \times 2$  matrix  $M$ , then it is very convenient to store its inversion or factorization in memory. This has been validated by the test case in the subsequent section.

### B. Solution by Modified Gaussian Elimination

The Gaussian elimination is always a competitive method for solving linear equations. Equation (17) is no exception. However, considering that matrix  $\mathbf{M}$  has constant off-diagonal elements, a modified Gaussian elimination method is provided that can reduce the computational complexity to a great extent. Assume that matrix  $\mathbf{M}$  has an initial structure shown as follows:

$$\mathbf{M} = [m_{i,j}]_{N \times N}$$

$$\begin{bmatrix} v_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,N-2} & c_{1,N-1} & c_{1,N} \\ c_{2,1} & v_{2,2} & c_{2,3} & \cdots & c_{2,N-2} & c_{2,N-1} & c_{2,N} \\ c_{3,1} & c_{3,2} & v_{3,3} & \cdots & c_{3,N-2} & c_{3,N-1} & c_{3,N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ c_{N-2,1} & c_{N-2,2} & c_{N-2,3} & \cdots & v_{N-2,N-2} & c_{N-2,N-1} & c_{N-2,N} \\ c_{N-1,1} & c_{N-1,2} & c_{N-1,3} & \cdots & c_{N-1,N-1} & v_{N-1,N-1} & c_{N-1,N} \\ c_{N,1} & c_{N,2} & c_{N,3} & \cdots & c_{N,N-2} & c_{N,N-1} & v_{N,N} \end{bmatrix} \quad (23)$$

where  $v$  represents the variable, which has two possible values and  $c$  represents the constant. Unlike the traditional Gaussian elimination, the first step of the modified method produces zeros in the last column using row 1 elements. After Step 1, matrix  $\mathbf{M}$  becomes

$$\mathbf{M}^{(1)} = [m_{i,j}^{(1)}]_{N \times N}$$

$$\begin{bmatrix} v_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,N-2} & c_{1,N-1} & c_{1,N} \\ v_{2,1}^{(1)} & v_{2,2}^{(1)} & c_{2,3}^{(1)} & \cdots & c_{2,N-2}^{(1)} & c_{2,N-1}^{(1)} & 0 \\ v_{3,1}^{(1)} & c_{3,2}^{(1)} & v_{3,3}^{(1)} & \cdots & c_{3,N-2}^{(1)} & c_{3,N-1}^{(1)} & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ v_{N-2,1}^{(1)} & c_{N-2,2}^{(1)} & c_{N-2,3}^{(1)} & \cdots & v_{N-2,N-2}^{(1)} & c_{N-2,N-1}^{(1)} & 0 \\ v_{N-1,1}^{(1)} & c_{N-1,2}^{(1)} & c_{N-1,3}^{(1)} & \cdots & c_{N-1,N-1}^{(1)} & v_{N-1,N-1}^{(1)} & 0 \\ v_{N,1}^{(1)} & v_{N,2}^{(1)} & v_{N,3}^{(1)} & \cdots & v_{N,N-2}^{(1)} & v_{N,N-1}^{(1)} & 0 \end{bmatrix} \quad (24)$$

where  $m_{i,j}^{(1)} = m_{i,j} - m_{i,N}m_{1,j}/m_{1,N}$ , for  $2 \leq i \leq N, 1 \leq j \leq N$ . The constants in the first column and the last row become variables because of Step 1 elimination while the other constants are still constants but with another value.

Similarly, Step 2 produces zeros in the second last column using row 2 elements and matrix  $\mathbf{M}^{(1)}$  becomes

$$\mathbf{M}^{(2)} = [m_{i,j}^{(2)}]_{N \times N}$$

$$\begin{bmatrix} v_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,N-2} & c_{1,N-1} & c_{1,N} \\ v_{2,1}^{(1)} & v_{2,2}^{(1)} & c_{2,3}^{(1)} & \cdots & c_{2,N-2}^{(1)} & c_{2,N-1}^{(1)} & 0 \\ v_{3,1}^{(2)} & v_{3,2}^{(2)} & v_{3,3}^{(2)} & \cdots & c_{3,N-2}^{(2)} & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ v_{N-2,1}^{(2)} & v_{N-2,2}^{(2)} & c_{N-2,3}^{(2)} & \cdots & v_{N-2,N-2}^{(2)} & 0 & 0 \\ v_{N-1,1}^{(2)} & v_{N-1,2}^{(2)} & v_{N-1,3}^{(2)} & \cdots & v_{N-1,N-1}^{(2)} & 0 & 0 \\ v_{N,1}^{(2)} & v_{N,2}^{(2)} & v_{N,3}^{(2)} & \cdots & v_{N,N-2}^{(2)} & 0 & 0 \end{bmatrix} \quad (25)$$

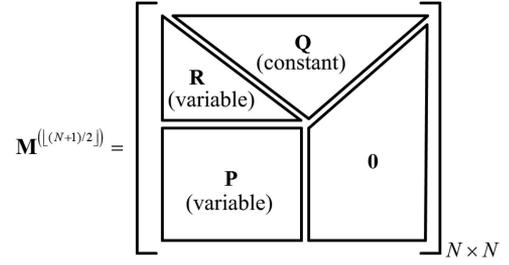


Fig. 8. Structure of  $\mathbf{M}$  after  $\lfloor (N+1)/2 \rfloor$  steps.

The constants in the second column and the second last row become variables while other constants remain constant. After  $\lfloor (N+1)/2 \rfloor$  steps, the structure of matrix  $\mathbf{M}$  will be like the one shown in Fig. 8, where  $\mathbf{P}$  is a  $\lfloor (N+1)/2 \rfloor \times \lfloor (N+1)/2 \rfloor$  variable matrix. Since the right-half block of  $\mathbf{M}^{(\lfloor (N+1)/2 \rfloor)}$  corresponding to  $\mathbf{P}$  is all zeros, solving  $\mathbf{P}$  using the traditional Gaussian elimination can obtain the first  $\lfloor (N+1)/2 \rfloor$  unknowns of  $\mathbf{I}$ . After this, substituting them into  $\mathbf{Q}$  and  $\mathbf{R}$ , the other unknowns of  $\mathbf{I}$  are also obtained. Note that the first  $\lfloor (N+1)/2 \rfloor$  steps of the modified Gaussian elimination is quite simple and can be conducted in advance. It can be proven that every element of matrix  $\mathbf{P}$  has at most four possible values, whereas the element of matrix  $\mathbf{R}$  has at most two possible values and matrix  $\mathbf{Q}$  is constant. So it will not take much memory to store them.

Because the dimension of  $\mathbf{P}$  is almost half as that of  $\mathbf{M}$  and the Gaussian elimination is a  $O(N^3/3)$  problem, if the number of nodes is approximately equal to the number of nonlinear elements, then the computation load corresponding to matrix  $\mathbf{P}$  is approximately one-eighth of that of the traditional method, i.e., solving (2) using a time-varying  $\mathbf{G}$ . Considering that the system may be partitioned into several subblocks due to the property discussed in Section IV-B, this number can be even smaller.

### C. Solution by Updating Inverse Using the Sherman–Morrison Formula

The Sherman–Morrison formula is a useful tool to update the inverse of an original matrix after a rank-1 modification [27]. Assume  $\mathbf{u}$  is a column vector and  $\mathbf{v}$  is a row vector, then the inverse of matrix  $\mathbf{M} + \mathbf{u}\mathbf{v}$  can be expressed as

$$(\mathbf{M} + \mathbf{u}\mathbf{v})^{-1} = \mathbf{M}^{-1} - \sigma \mathbf{M}^{-1} \mathbf{u}\mathbf{v}\mathbf{M}^{-1} \quad (26)$$

where  $\sigma$  is a scalar and  $\sigma = 1/(1 + \mathbf{v}\mathbf{M}^{-1}\mathbf{u})$ .

Considering that matrix  $\mathbf{M}$  has variables only on diagonal, any two possible variations of  $\mathbf{M}$  can be mutually transformed by a series of rank-1 modifications. The computational complexity of calculating the modified inverse  $\mathbf{M}_1^{-1}$  from a base matrix  $\mathbf{M}_0^{-1}$  is  $O(mN^2)$ , where  $m$  is the number of diagonals of  $\mathbf{M}_1$  that differ from those of  $\mathbf{M}_0$ . If there are enough base matrices that have been computed and prestored in memory so that  $m \leq \lfloor N/3 \rfloor$ , then this method will have advantage over the traditional Gaussian elimination method. Obviously, more base matrices reduce the computational complexity but require more memory space. It has to be pointed out that the actual diagonal combinations of  $\mathbf{M}$  is far less than  $2^N$ , for example,

$\mathbf{M}$  that consists of all voltage equations is most likely to be singular because all the power electronic switches cannot be switched ON simultaneously. This will significantly reduce the memory storage demand.

Note that formula (26) may cause numerical error when  $(1 + \mathbf{v}\mathbf{M}^{-1}\mathbf{u})$  is very small. This happens frequently because the diagonal elements of  $\mathbf{M}^{-1}$  are often very close to  $\pm 1$ . To the best of author's knowledge, when choosing the base matrix  $\mathbf{M}_0$ , it is better to select certain rows and add an additional 1 to the diagonal elements of current equation and subtract an additional 1 from the diagonal elements of voltage equation and leave the additional modified diagonal elements to be the last steps of the Sherman–Morrison update process. This will avoid the numerical distortion in most cases.

The above-mentioned three methods provide three perspectives to solve diagonal time-varying linear equations. However, they each have their own advantages and eligible applications. When solving a new system, decoupling is always the first consideration. After the decoupling step, a proper method needs to be chosen. Generally speaking, precomputed inversion or factorization is suitable for small size  $\mathbf{M}$ ; the Sherman–Morrison method is suitable for medium size; and the Gaussian elimination is suitable for large size. In addition, the Gaussian elimination is more suitable for sequential programs (such as CPU-based program) because of low computational complexity, whereas the Sherman–Morrison method is more suitable for parallel programs (such as FPGA-based program) because the intermediate quantities do not depend on each other; thus in each process, the  $N^2$  updating values can be calculated simultaneously, which makes the  $O(mN^2)$  problem  $O(m)$  with respect to computation time.

Another important feature that has to be pointed out is that this method is always numerically stable, no matter what the nonlinearities are and how they connect with each other. This is because the equivalent representation of all the nonlinear elements has an internal resistor. Thus, from the system's point of view, it looks like to deal with a pure resistor network, and the only time-varying part is the companion current source and the system itself does not have any instability risk.

## V. OFFLINE SIMULATION VALIDATION

The MMC and NPC converters are two commonly used topologies in power electronic industry and will be used as two test cases to validate the proposed simulation method. The output waveforms and computation efficiency are compared with the existing system-level and device-level simulation tools—PSCAD/EMTDC and SaberRD.

### A. Offline Simulation Results of MMC

The MMC topology is an effective configuration to achieve high voltage and power using moderate switching devices. It is widely used in the high voltage direct current (HVDC) transmission systems around the world. For a long time, simulation of MMC topology behavior was a tough task because it has too many semiconductor switches in one converter. But using the

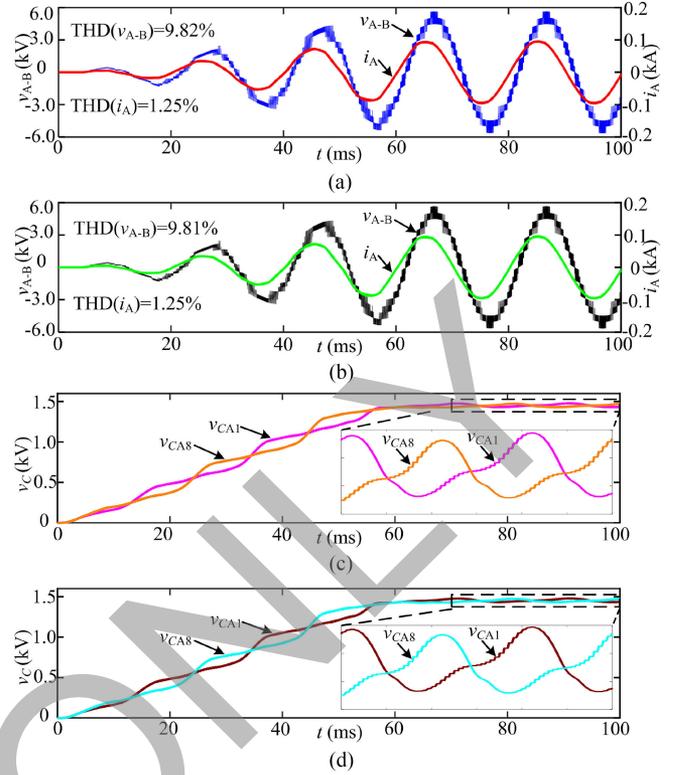


Fig. 9. Comparison of five-level MMC simulation waveforms between PSCAD/EMTDC and the proposed method based program. (a) Line-line voltage ( $v_{A-B}$ ) and load current ( $i_A$ ) waveforms from PSCAD/EMTDC. (b) Counterparts of (a) from the proposed method based program. (c) SM capacitor voltage ( $V_{CA1}$  and  $V_{CA8}$ ) waveforms from PSCAD/EMTDC. (d) Counterparts of (c) from the proposed method based program.

proposed method, this task becomes comparatively easy because all the SMs in MMC are decoupled from each other in nature.

A three-phase five-level MMC is constructed and served as the test case and its parameters are listed in Table V (see the appendix). The resulting waveforms of this five-level MMC from PSCAD/EMTDC and a C-program employing the proposed method are shown in Fig. 9. The companion current source of nonlinear elements are solved by precomputed inversion, as discussed in Section IV-A. The modulation scheme adopted here is the phase shift pulsewidth modulation (PWM) with a displacement angle of  $45^\circ$  between upper and lower arms [28], [29]. As for the capacitor voltage, a combination of averaging and individual proportional–integral (PI) controllers are designed to achieve voltage balancing (see reference [30]). The dc-link voltage is set to soft start from 0 to rated value in 50 ms and the simulation time step is 500 ns. Differences in these waveforms can barely be observed. Further calculations show that the total harmonic distortion (THD) of line-line voltage between A and B phases ( $v_{A-B}$ ) is 9.82% (PSCAD/EMTDC results) and 9.81% (the proposed method results), respectively, and the THD of the load current of phase A ( $i_A$ ) is 1.25% for both the results. Two capacitor voltages, namely the upper most and lower most SM of phase A leg ( $C_{A1}$  and  $C_{A8}$ ), are selected to be drawn in Fig. 9. The mean

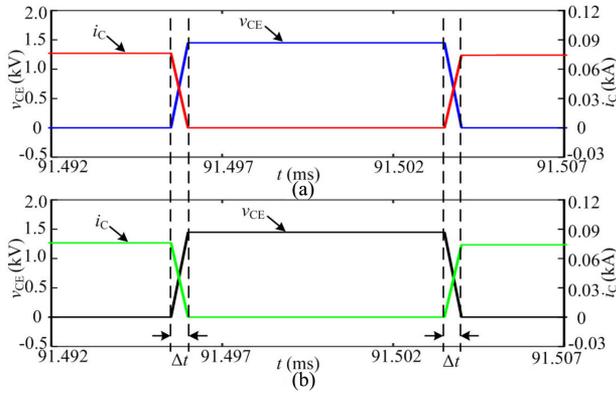


Fig. 10. Comparison of switching ON and OFF waveforms between PSCAD/EMTDC and the proposed method based program. (a) PSCAD/EMTDC results. (b) Proposed method based program results.

square error of  $v_{CA1}$  between these two simulation tools is  $9.8189 \times 10^{-7}$  and  $9.8398 \times 10^{-7}$  for  $v_{CA8}$ . These values indicate that the output waveforms from PSCAD/EMTDC and the proposed method based program are very close to each other.

The nonlinear element, i.e., the insulated-gate bipolar transistor (IGBT), is another major concern of the simulation accuracy. In PSCAD/EMTDC, an IGBT is modeled as a two-state resistance ( $R_{on}$  and  $R_{off}$ ) and the switching ON and OFF waveforms of an IGBT are shown in Fig. 10(a). As can be seen, in ON state, the voltage across the IGBT's collector and emitter ( $v_{CE}$ ) is always approximately to zero, whereas in OFF state, the IGBT's collector current ( $i_C$ ) is always approximately to zero and the switching transition between ON/OFF states is accomplished in one time step. As a comparison, the proposed method based program also applies a simple switch model to imitate the ideal switch, namely treat the switch as a voltage source whose value is zero in ON state and a current source whose value is zero in OFF state and no switching transient is assumed. The resulting waveforms are shown in Fig. 10(b). The nonlinear element behaviors from these two simulation tools are also very close and they both resemble the ideal switch.

The computational efficiency may be the most significant difference between these two tools. Execution time test is conducted between PSCAD/EMTDC and the C-program. Because comparison between a general purpose software and a specific C-program is not particularly fair, the following efforts have been made to make it fairer.

- 1) The compile process of the program is not included and only the execution time is accounted for.
- 2) When running the PSCAD/EMTDC model, no graphic output is added so as to avoid extra time for drawing figures.
- 3) The execution time for the C-program is measured in debug mode rather than in release mode.

To conduct a 100-ms simulation using time step of 500 ns on a computer with Intel i7-3770 processor, PSCAD/EMTDC spends around 34 s, whereas the proposed method based program only costs 3.361 s, which is approximately ten times faster.

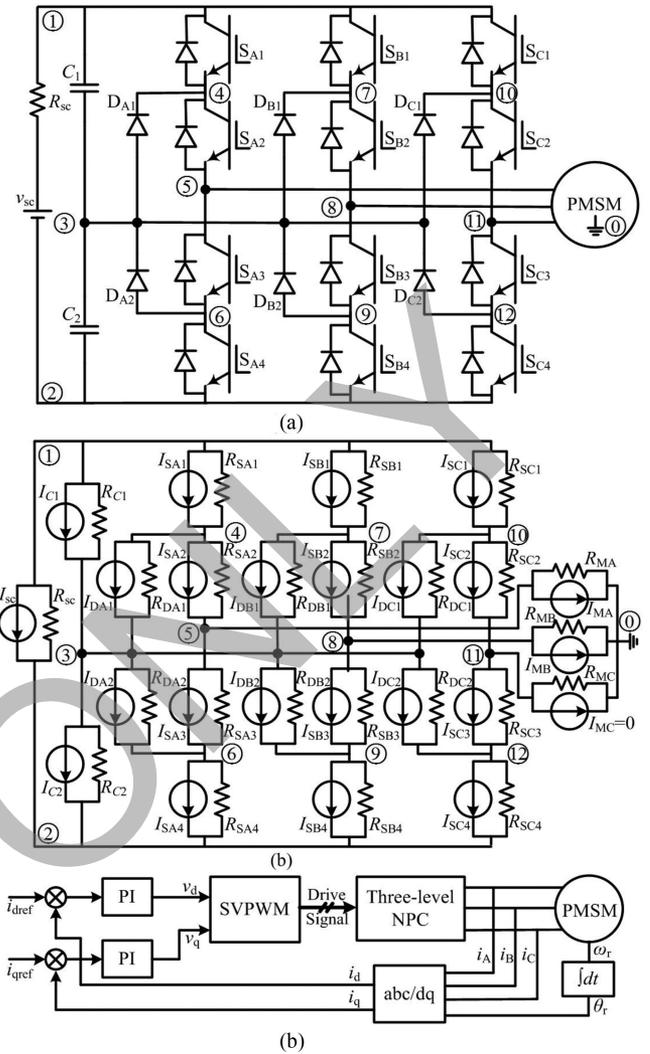


Fig. 11. Configuration of the three-level NPC converter fed PMSM drive system. (a) Topology structure. (b) Equivalent circuit. (c) Control scheme.

This demonstrates that the proposed method has great advantage over the traditional method in terms of computational efficiency while maintaining almost the same accuracy. As the system includes more nonlinear elements, this advantage will become more significant.

### B. Offline Simulation Results of NPC Topology

The NPC topology is another commonly used multilevel converter that is often applied in medium power circumstances and the three-level NPC converter is the most mature one. In this test case, a three-level NPC converter fed permanent magnet synchronous machine (PMSM) drive system is constructed. The configuration of the test setup and its equivalent circuit as well as the control scheme are illustrated in Fig. 11.

There are 12 nodes, 3 linear elements (1 source and 2 capacitors), and 19 nonlinear elements (12 IGBTs, 6 diodes, and 1 machine) in this system. Because the machine is a three-terminal element, it is represented by three resistors and the

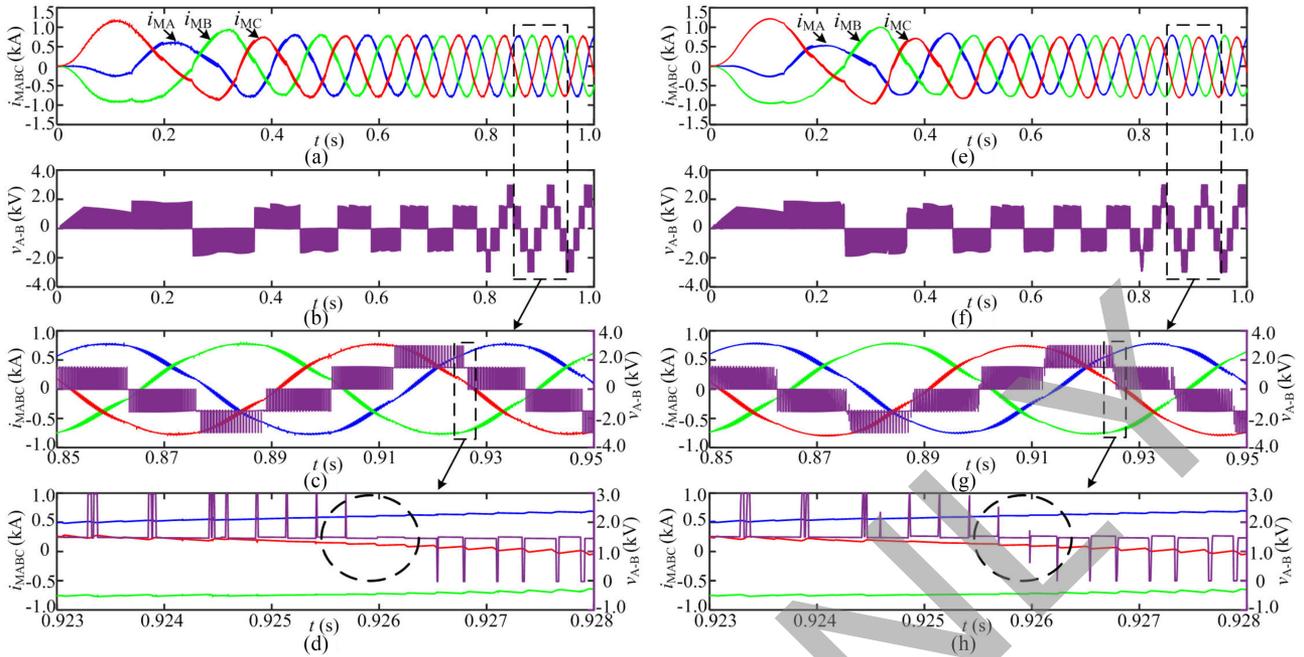


Fig. 12. Comparison of three-level NPC-fed PMSM drive system start-up waveforms between PSCAD and the proposed method based program. (a) PMSM three-phase stator current from PSCAD/EMTDC. (b) NPC line-line voltage between phases A and B from PSCAD/EMTDC. (c) Magnified waveforms of (a) and (b). (d) Magnified waveforms of (c). (e)–(h) Counterparts of (a)–(d) from the proposed method based program.

corresponding companion current sources. However, the three-phase currents of the machine are not independent, their sum is always zero, which means there are only two degrees of freedom, thus one companion current source ( $I_{MC}$ ) can be set to zero. Then, matrix  $\mathbf{A}$  is of  $12 \times 23$  and  $\mathbf{M}$  is of  $20 \times 20$ . Among the 20 nonlinear elements, the IGBTs and diodes will transfer types during simulation while the machine will always be treated as a current source. This means, as in (14),  $\mathbf{M}$  can be partitioned into a  $18 \times 18 \mathbf{M}_{11}$  and a  $2 \times 2 \mathbf{M}_{22}$ . When applying matrix transformation, as shown in (16), one would find that the  $18 \times 18$  diagonal time-varying matrix is transformed into three identical  $6 \times 6$  decoupling matrices. Each matrix represents one phase leg of the three-level NPC converter. It makes sense because the three-phase legs connect in parallel with the capacitor  $C_1$  and  $C_2$  whose equivalent resistor is very small. As explained in Section IV, a small resistor will decouple the subsystems that connect in parallel with it. The decoupling nature of the nonlinear elements significantly reduces the computational burden. In this test case, the time-varying  $6 \times 6$  matrix equations is solved by the modified Gaussian elimination method discussed in Section IV-B. Note that node 0 in Fig. 11 is a virtual node that has no physical meaning. It only serves as a reference point and even if the stator windings of PMSM connect in wye type, the potential of node 0 is not equal to that of the neutral point of the wye configuration.

The parameters of the system are listed in Table VI (see the appendix). The PMSM model applied here is a fifth-order model that has three stator windings and two short-circuited damping windings.

The output simulation waveforms of PSCAD/EMTDC and the proposed method based C-program at the time step of 200 ns

are shown in Fig. 12. The machine is set to start up at half load and the closed-loop space vector pulsewidth modulation (SVPWM) control strategy is employed where the  $d$ -axis current is tuned to zero and the  $q$ -axis current is tuned to 1 kA. The differences of waveforms from these two simulation tools can hardly be distinguished. Only in very small time-scale scope, one can find the slight discrepancies, as denoted in Fig. 12(d) and (h). This is because in PSCAD/EMTDC, the switch devices are simple  $R_{on}/R_{off}$  model, whereas in the new method based program, the realistic IGBT and diode switching characteristics are considered. Actually, the waveforms in Fig. 12(h) are more close to experimental results (see [31, Fig. 8] and [32, Fig. 9]). There are always voltage spikes during the voltage-level transition and they are indeed caused by the switching transients of semiconductor devices.

To further validate the accuracy of the proposed method based program, the IGBT and diode switching transient waveforms in NPC topology are also recorded and compared with SaberRD that has very sophisticated semiconductor models. The target IGBT and diode part number employed in simulation are FZ1500R33HL3 from Infineon and VS-SD1053CS20L from Vishay. The switching parameters of these two devices from datasheet are listed in Table II.

The outcome switching transients from these two simulation tools are illustrated in Fig. 13 and the corresponding switching parameters are also shown in Table II. As can be seen, the switching transient processes are highly nonlinear and can take several microseconds. Both these two simulation tools reproduced the switching transients properly. The key parameters ( $t_{d(on)}$ ,  $t_r$ ,  $t_{d(off)}$ ,  $t_f$ ,  $t_{rr}$ ) are very close to that given in datasheet. In this sense, the proposed method based

TABLE II  
SWITCHING PARAMETERS OF IGBT AND DIODE

Parameter	Datasheet	SaberRD®	Proposed method based program	
IGBT	Turn-on delay time ( $t_{d(on)}$ )	0.50 $\mu$ s	0.52 $\mu$ s	0.54 $\mu$ s
	Rise time ( $t_r$ )	0.55 $\mu$ s	0.53 $\mu$ s	0.58 $\mu$ s
	Turn-off delay time ( $t_{d(off)}$ )	4.3 $\mu$ s	4.3 $\mu$ s	4.4 $\mu$ s
	Fall time ( $t_f$ )	0.40 $\mu$ s	0.38 $\mu$ s	0.42 $\mu$ s
Diode	Reverse recovery time ( $t_{rr}$ )	4.0 $\mu$ s	4.0 $\mu$ s	4.0 $\mu$ s

program achieved comparable switching transients accuracy as with SaberRD.

Again, the computational efficiency performance of these simulation tools differ greatly. To conduct a 1-s simulation of this NPC-fed PMSM drive system at the time step of 200 ns on an Intel i7-3770 processor computer platform, PSCAD/EMTDC spends around 143 s, whereas the proposed method based program only consumes 13.288 s—more than ten times faster. As for the SaberRD, it takes several hours to finish the same test set-up due to the complexity of its iterative algorithms and it utilizes the variable time step [33], thus its computational efficiency is not compared here.

The above-mentioned two test cases demonstrate that the method proposed in this paper is feasible in offline simulation. It can give almost the same accuracy results of mature simulation software while spending far less execution time. In addition, this method has high generality. It can be applied to deal with most of (if not all) the nonlinearities in the major power electronic circuit topologies while retaining its high computational efficiency.

## VI. REAL-TIME SIMULATION VALIDATION

Real-time simulation is widely accepted by industrial engineers and designers as it has the ability to offer hardware-in-the-loop tests that are more close to real applications compared with offline simulation. In real-time simulation, the demand for computational efficiency is extremely high because it has to update the states of emulated system within every time step. In most cases, real-time simulation is very hard to achieve and simplifications have to be made to trade accuracy for speed. However, the method proposed in this paper lowers the computational burden while still retaining almost the same accuracy with the traditional method. This means it is very suitable for real-time simulation. In addition, the constant admittance matrix exploits the advantage of parallel computing to its full strength as every node voltage can be calculated independently and simultaneously.

In this section, the above-mentioned two test cases are implemented in real time on the Xilinx Virtex UltraScale+VCU118-ES1 FPGA evaluation platform. The FPGA-based real-time simulator has merits such as low cost, low hardware utilization, easy maintenance, easy reprogramming, and high reliability. Because of the good generality of the proposed method, there is no special treatment of the circuit model itself like many other

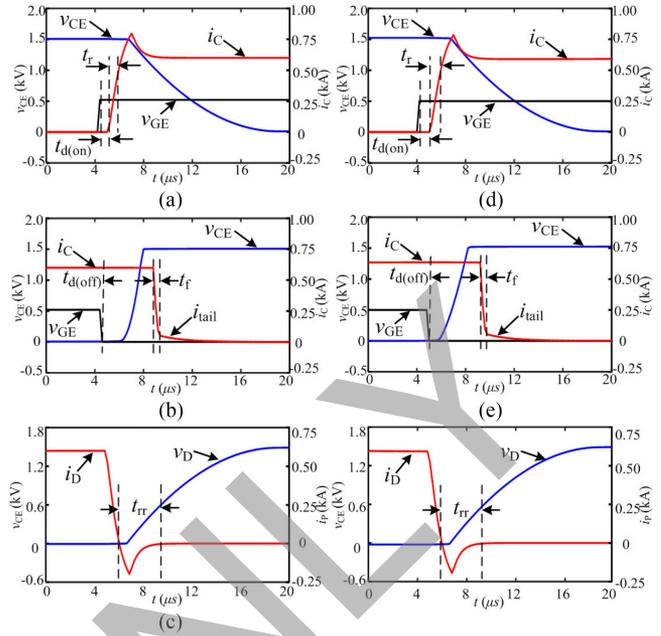


Fig. 13. Comparison of switching transients between SaberRD and new method based program. (a) IGBT turn-ON transients from SaberRD. (b) IGBT turn-OFF transients from SaberRD. (c) Diode reverse recovery transients from SaberRD. (d)–(f) Counterparts of (a)–(c) from the proposed method based program.

real-time simulation programs. Thus, the signal flowchart for these two test models is common and is given in Fig. 14. The whole project is divided into two parallel working modules: the control and the model module. The control module is designed to generate driving signals of semiconductor devices. Some of the common algorithms, such as coordinates transformation, PI controller, SVPWM, and PWM generator, are implemented here. The model module is responsible for updating the states of the simulated system and is partitioned into two sequential tasks: calculating the node voltages and calculating the companion current sources. The node voltages can be obtained by direct calculation using (6) or through lower-upper (LU) factorization of (2). In the FPGA-based program, the former is preferred although solving (2) has higher precision because (6) is more suitable for parallel programming and thus saves execution time; however, it is not always the case as it depends on the programmer and system sizing.

Calculation of the companion current sources is the key feature of this new method. It is partitioned into two successive processes—updating the states of linear and nonlinear elements. The linear elements will be dealt with first either by the trapezoidal rule or the backward Euler rule or any other integration method. Trapezoidal rule is always stable and has an error term of  $O((\Delta t)^3)$ , whereas backward Euler rule has simpler form but lower precision. The updating of nonlinear elements starts with judgment of them to be treated as current or voltage sources. The judgment condition can be obtained from control signals or system states. After this step, the exact value of the equivalent current or voltage source is calculated in accordance with the nonlinear functions and a new matrix  $M$  is developed accordingly. Finally, the diagonal

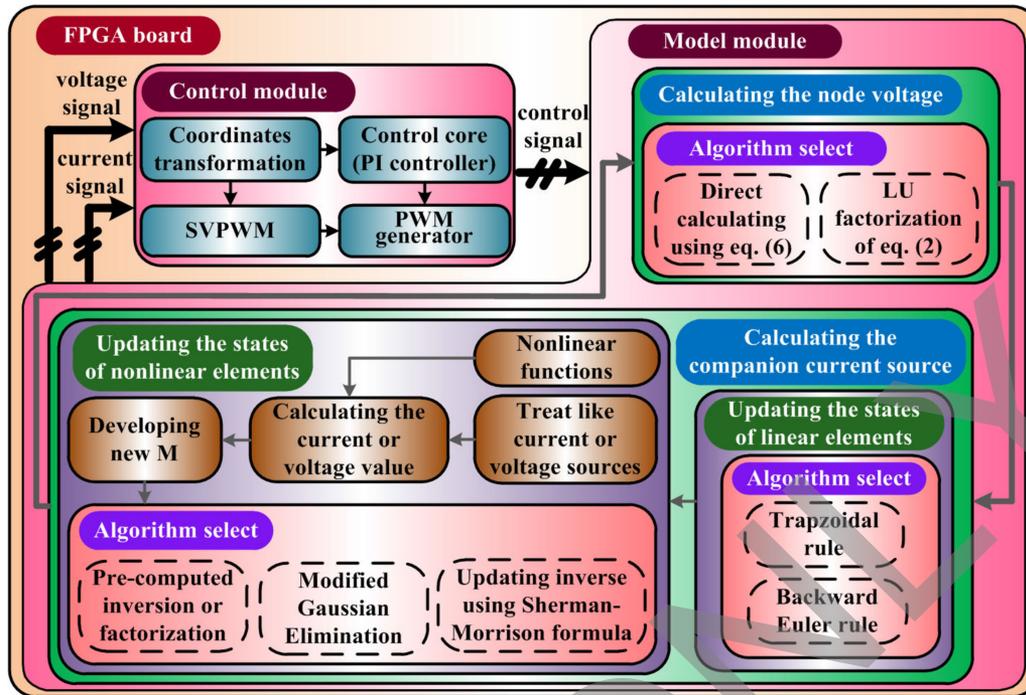


Fig. 14. Signal flowchart of the proposed method implemented on FPGA.

time-varying matrix equations is solved by either precomputed inversion or factorization or modified Gaussian elimination or updating inverse using the Sherman–Morrison formula. The selection of algorithm depends on many factors, especially the system sizing as discussed in Section IV. Considering that the program is implemented on the FPGA board, the first and third methods will have higher priority. In the two test cases that are considered here, the MMC model will use precomputed inversion and the NPC-fed PMSM drive model will employ the updating inverse by the Sherman–Morrison formula.

#### A. Real-Time Simulation Results of MMC

As discussed in Section IV, the matrix  $M$  in the MMC model is of  $2 \times 2$  and has only two possible variations corresponding to upper switch ON lower switch OFF mode and upper switch OFF lower switch ON mode, respectively. Therefore, the process of updating the states of nonlinear elements is easy as these two possible  $M$  matrices are precomputed and stored in memory. The only challenge comes from calculating the node voltages. Because the MMC topology has many switches, the matrix  $A$  in (6) has a size of  $53 \times 82$  for a five-level topology. If they are implemented fully parallel in FPGA, more than 4000 multipliers are needed and each multiplier will consume several digital signal processor (DSP) blocks that is unbearable for the existing FPGA board. To reduce the hardware consumption, several techniques are adopted here.

1) The data format employed in this test case is a 59-b fixed point number to guarantee lower DSP blocks consumption and efficient precision as well. Although the  $59 \times 59$  fixed point multiplication has to use ten DSP blocks that is higher than double-precision floating point format, the addition of fixed

TABLE III  
HARDWARE UTILIZATION OF THE FIVE-LEVEL MMC MODEL

Resource	LUT	FF	BRAM	DSP
Utilization	582115 (49.24%)	211466 (8.94%)	37.5 (1.74%)	2140 (31.3%)

LUT-look-up table; FF-flip-flops; BRAM-block random access memory

point numbers is much simpler than floating point numbers. Thus, the overall hardware consumption is reduced.

2) The calculation process of each node voltage that involves 82 multiplications and 81 additions is partitioned into 4 groups to deal with 19, 20, 21, and 22 multiplications, respectively. Each group is instantiated by a multiply-accumulate module. This means that each multiplier is reused about 20 times, and more importantly, the pipeline technique is included.

With the help of these techniques, the 500 ns time-step real-time simulation of the five-level MMC test case discussed in Section V-A is realized. The system clock implemented is 100 MHz, which means that in every time step, the state updating is accomplished within 50 system clocks. The hardware utilization and real-time implementation waveforms are illustrated in Table III and Fig. 15, respectively. The results are almost the same with offline waveforms except they are real time.

#### B. Real-Time Simulation Results of NPC Topology

The NPC-fed PMSM drive system experienced a much simpler node voltage calculation process as its system size is much smaller. What should be paid attention to is solving the diagonal time-varying matrix equations. As analyzed in

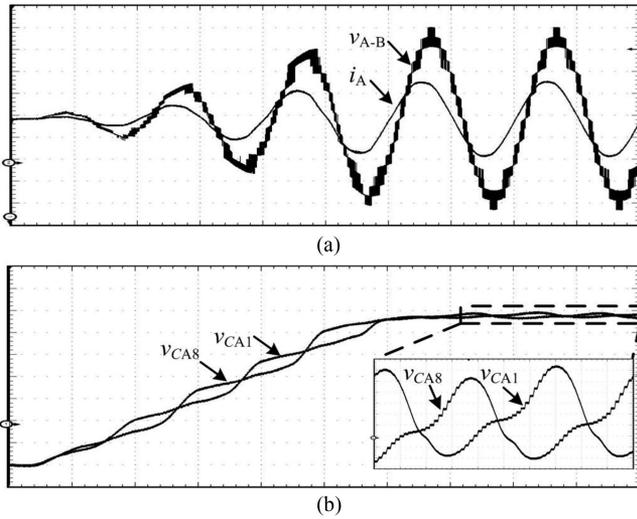


Fig. 15. Real-time implementation waveforms of the five-level MMC system. (a) Ch1: Line-line voltage ( $V_{A-B}$ ), y-axis: 1.5 kV/div. Ch2: load current ( $i_A$ ) waveforms, y-axis: 0.5 kA/div, x-axis: 10 ms/div. (b) SM capacitor voltage (Ch1:  $V_{CA1}$  and Ch2:  $V_{CA8}$ ) waveforms, y-axis: 0.2143 kV/div, x-axis: 10 ms/div.

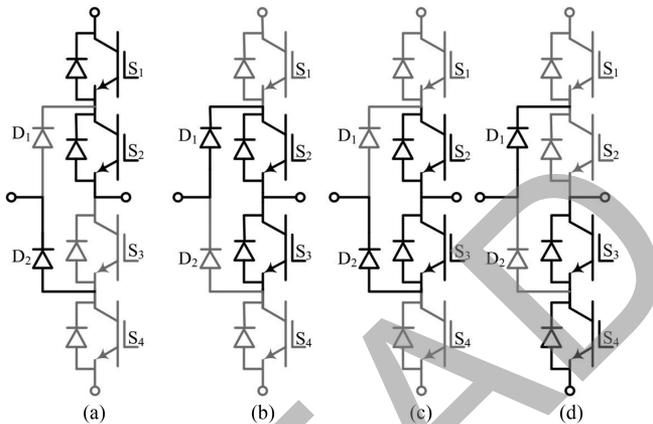


Fig. 16. Four possible switching combinations of three-level NPC phase leg. Dark color represents switch-ON, whereas gray color represents switch-OFF. (a) Output positive voltage level. (b) Output neutral voltage level and load current are positive. (c) Output neutral voltage level and load current are negative. (d) Output negative voltage level.

TABLE IV  
HARDWARE UTILIZATION OF THE THREE-LEVEL NPC MODEL

Resource	LUT	FF	BRAM	DSP
Utilization	125418 (10.61%)	98601 (4.17%)	106 (4.91%)	998 (14.59%)

LUT-look-up table; FF-flip-flops; BRAM-block random access memory

Section V-B,  $\mathbf{M}$  is a  $6 \times 6$  matrix, but the possible variations of  $\mathbf{M}$  are much smaller than  $2^6$ ; actually there are only four variations representing four possible switching combinations of one phase leg, as indicated in Fig. 16. Note that there is no need to worry about the switching pair deadtime as the dead-time modes are already included in the above-mentioned four possible modes.

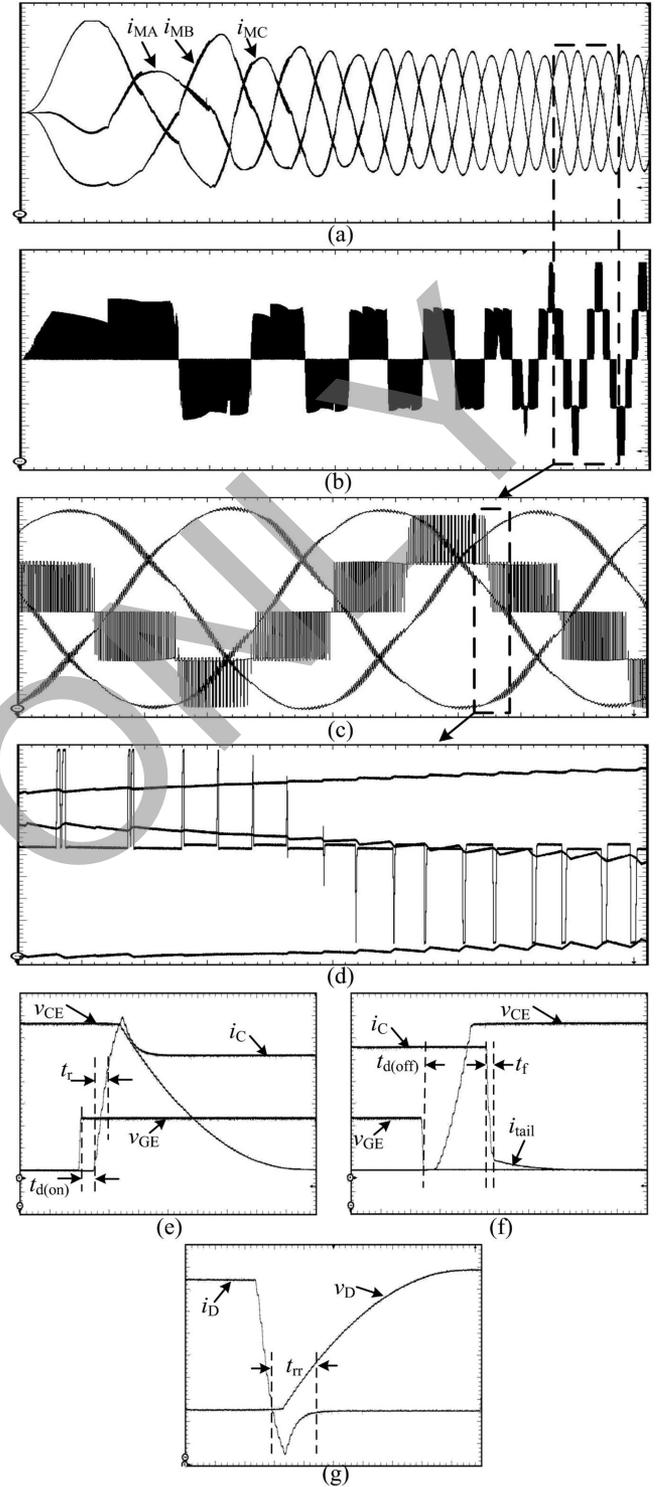


Fig. 17. Real-time implementation waveforms of the three-level NPC fed-PMSM system. (a) Ch1: PMSM phase A current ( $i_{MA}$ ), Ch2: PMSM phase B current ( $i_{MB}$ ), Ch3: PMSM phase C current ( $i_{MC}$ ), y-axis: 0.3 kA/div, x-axis: 0.1 s/div. (b) NPC line-line voltage between phase A and B, y-axis: 0.7143 kV/div, x-axis: 0.1 s/div. (c) Magnified waveforms of (a) and (b), Ch1-Ch3: y-axis: 0.18 kA/div, Ch4: y-axis: 0.7143 kV/div, x-axis: 0.01 s/div. (d) Magnified waveforms of (c), Ch1-Ch3: y-axis: 0.17 kA/div, Ch4: y-axis: 0.33 kV/div, x-axis: 0.5 ms/div. (e) and (f) IGBT turn-ON and turn-OFF transients, Ch1:  $V_{CE}$ , y-axis: 0.25 kV/div, Ch2:  $i_C$ , y-axis: 0.125 kA/div, Ch3:  $V_{GE}$ , y-axis: 5 V/div, x-axis: 2  $\mu$ s/div. (g) Diode reverse recovery transients, Ch1: y-axis: 0.1 kA/div, Ch2: 0.24 kV/div x-axis: 2  $\mu$ s/div.

Observing the four possible switching combinations, modes (a) and (c) have four same switches ( $S_2$ ,  $S_4$ ,  $D_1$ , and  $D_2$ ) and two different switches ( $S_1$  and  $S_3$ ), and modes (b) and (d) have four same switches ( $S_1$ ,  $S_3$ ,  $D_1$ , and  $D_2$ ) and two different switches ( $S_2$  and  $S_4$ ). Thus modes (b) and (c) can be served as the base matrices in the Sherman–Morrison formula updating process. From mode (c) to mode (a) and from mode (b) to mode (d) only takes two updating processes. Furthermore, in order to accelerate the renewal of system states in every time step, the Sherman–Morrison updating process is not included in the main sequential computation loop of system states in the model module, but rather works in parallel with the main loop just like the control module. This means, whenever there is a new mode coming, the control module will be responsible for updating the new inverse of matrix  $M$ . It is when the new inverse of matrix  $M$  is ready that the new mode will take effect. This kind of operation will introduce some delay in the control loop, however, considering that there is always real delay (approximately half the carrier period of switching devices) in power electronic converter control and the introduced Sherman–Morrison updating process delay is less than  $1 \mu\text{s}$ , which is much smaller than half the carrier period, the resulting effect is negligible. In this arrangement, the NPC-fed PMSM drive system achieved 200 ns time step on the 100-MHz system clock FPGA board. The resulting hardware utilization is summarized in Table IV and the real-time implementation waveforms are shown in Fig. 17.

## VII. CONCLUSION

This paper provided a new perspective to deal with nonlinearities in a fast and stable manner in power electronic circuit simulation. The proposed method utilizes a uniform representation to substitute the nonlinearities in order to keep the system admittance matrix constant. The nonlinear elements then can be further viewed as current or voltage sources according to their nature or system states and the corresponding current or voltage equations are developed accordingly. Therefore, the computational difficulty is transformed to solve a diagonal time-varying matrix equation. Three methods are provided to solve it, which can cover most circumstances that are encountered in practical applications. The offline and real-time test cases validated the feasibility of this new method and also demonstrated its fast simulation speed. In addition, the proposed method has very good generality: it can be used to deal with most nonlinearities in major power electronic topologies while achieving comparable accuracy as with the mature simulation software at both the system level and device level. This method can be a good alternative algorithm to accelerate the speed in power electronic circuit simulation and it is also suitable to be made into a drag and drop tool incorporated with existing commercial softwares.

## APPENDIX

See Tables V and VI.

TABLE V  
PARAMETERS OF THE FIVE-LEVEL MMC

Parameter	Value
No. of SMs per Arm	4
Rated DC-link Voltage	6 kV
Arm Inductance	3 mH
SM Capacitance	1900 $\mu\text{F}$
Carrier Frequency	2000 Hz
Reference Frequency	50 Hz
Load Inductance	6 mH
Load Resistance	30 $\Omega$

TABLE VI  
PARAMETERS OF THE THREE-LEVEL NPC AND PMSM

	Parameter	Value
NPC	Rated DC Bus Voltage	3 kV
	DC source resistor ( $R_{sc}$ )	0.01 $\Omega$
	DC Bus Capacitor ( $C_1$ , $C_2$ )	10 mF
	Carrier Frequency	1800 Hz
PMSM	Rated MVA	3.65 MVA
	Rated Voltage (L-L)	4 kV
	Rated Frequency	60 Hz
	Stator Winding Resistance	0.011 p.u.
	Stator Leakage Reactance	0.064 p.u.
	d-axis Reactance ( $X_d$ )	0.689 p.u.
	q-axis Reactance ( $X_q$ )	0.689 p.u.
	d-axis Damper Winding Resistance ( $R_{kd}$ )	0.055 p.u.
	d-axis Damper Winding Reactance ( $X_{kd}$ )	0.62 p.u.
	q-axis Damper Winding Resistance ( $R_{kq}$ )	0.183 p.u.
	q-axis Damper Winding Reactance ( $X_{kq}$ )	1.175 p.u.
	Magnetic Strength ( $R_{kq}$ )	1.0 p.u.
Inertia Constant ( $H$ )	1 s	
Friction Factor ( $B$ )	0.0005	
Load Torque ( $T_L$ )	0.5 p.u.	

## REFERENCES

- [1] 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.
- [2] H. W. Dommel, "Digital computer solution of electromagnetic transients in single- and multiphase networks," *IEEE Trans. Power App. Syst.*, vol. PAS-88, no. 4, pp. 388–399, Apr. 1969.
- [3] H. W. Dommel, *Electro-Magnetic Transients Program (EMTP) Theory Book*. Vancouver, BC, Canada: Microtran Power Syst. Anal. Corp., 1996.
- [4] T. Zhao, Q. Li, and J. Qian, "Investigation on digital algorithm for on-line monitoring and diagnostics of metal oxide surge arrester based on an accurate model," *IEEE Trans. Power Del.*, vol. 20, no. 2, pp. 751–756, Apr. 2005.
- [5] M. Armstrong, J. R. Marti, L. R. Linares, and P. Kundur, "Multilevel MATE for efficient simultaneous solution of control systems and nonlinearities in the OVNI simulator," *IEEE Trans. Power Syst.*, vol. 21, no. 3, pp. 1250–1259, Aug. 2006.
- [6] W. Wang, Z. Shen and V. Dinavahi, "Physics-based device-level power electronic circuit hardware emulation on FPGA," *IEEE Trans. Ind. Inform.*, vol. 10, no. 4, pp. 2166–2179, Nov. 2014.
- [7] T. Noda and T. Kikuma, "A robust and efficient iterative scheme for the EMT simulations of nonlinear circuits," *IEEE Trans. Power Del.*, vol. 26, no. 2, pp. 1030–1038, Apr. 2011.
- [8] Y. Chen and V. Dinavahi, "An iterative real-time nonlinear electromagnetic transient solver on FPGA," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2547–2555, Jun. 2011.

- [9] F. Bizzarri, A. Brambilla, G. S. Gajani, and S. Banerjee, "Simulation of real world circuits: extending conventional analysis methods to circuits described by heterogeneous languages," *IEEE Circuits Syst. Mag.*, vol. 14, no. 4, pp. 51–70, Oct.–Dec. 2014.
- [10] *EMTDC User's Guide Version 4.2.0.*, Manitoba HVDC Res. Center, Winnipeg, MB, Canada, 2005.
- [11] A. Myaing and V. Dinavahi, "FPGA-based real-time emulation of power electronic systems with detailed representation of device characteristics," *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 358–368, Jan. 2011.
- [12] A. M. Gole, R. W. Menzies, H. M. Turanli, and D. A. Woodford, "Improved interfacing of electrical machine models to electromagnetic transients programs," *IEEE Trans. Power App. Syst.*, vol. PAS-103, no. 9, pp. 2446–2451, Sep. 1984.
- [13] W. Eberle, Z. Zhang, Y. F. Liu, and P. C. Sen, "A practical switching loss model for buck voltage regulators," *IEEE Trans. Ind. Electron.*, vol. 24, no. 3, pp. 700–713, Mar. 2009.
- [14] L. Qu and P. L. Chapman, "Extraction of low-order non-linear inductor models from a high-order physics-based representation," *IEEE Trans. Ind. Electron.*, vol. 21, no. 3, pp. 813–817, May 2006.
- [15] K. Strunz and E. Carlson, "Nested fast and simultaneous solution for time-domain simulation of integrative power-electric and electronic systems," *IEEE Trans. Power Del.*, vol. 22, no. 1, pp. 277–287, Jan. 2007.
- [16] F. Vasca, L. Iannelli, M. K. Camlibel, and R. Frasca, "A new perspective for modeling power electronics converters: Complementarity framework," *IEEE Trans. Ind. Electron.*, vol. 24, no. 2, pp. 456–468, Feb. 2009.
- [17] Z. Shen and V. Dinavahi, "Real-time device-level transient electrothermal model for modular multilevel converter on FPGA," *IEEE Trans. Power Electron.*, vol. 31, no. 9, pp. 6155–6168, Sep. 2016.
- [18] B. Gustavsen, "Wideband transformer modeling including core nonlinear effects," *IEEE Trans. Power Del.*, vol. 31, no. 1, pp. 219–227, Feb. 2016.
- [19] S. Y. R. Hui and S. Morrall, "Generalized associated discrete circuit model for switching devices," *IEE Proc., Sci., Meas. Technol.*, vol. 141, no. 1, pp. 57–64, Jan. 1994.
- [20] Q. Mu, J. Liang, X. Zhou, Y. Li, and X. Zhang, "Improved ADC model of voltage-source converters in DC grids," *IEEE Trans. Power Electron.*, vol. 29, no. 11, pp. 5738–5748, Nov. 2014.
- [21] P. Pejović and D. Maksimović, "A new algorithm for simulation of power electronic systems using piecewise-linear device models," *IEEE Trans. Power Electron.*, vol. 10, no. 3, pp. 340–348, May, 1995.
- [22] A. Benigni, A. Monti, and R. A. Dougal, "Latency-based approach to the simulation of large power electronics systems," *IEEE Trans. Power Electron.*, vol. 29, no. 6, pp. 3201–3213, Jun. 2014.
- [23] S. Y. R. Hui, K. K. Fun, and C. Christopoulos, "Decoupled simulation of multi-stage power electronic systems using transmission-line links," in *PESC '92 Rec. 23rd Annu. IEEE Power Electron. Spec. Conf.*, Toledo, Spain, 1992, pp. 1324–1330.
- [24] B. Asghari and V. Dinavahi, "Novel transmission line modeling method for nonlinear permeance network based simulation of induction machines," *IEEE Trans. Magn.*, vol. 47, no. 8, pp. 2100–2108, Aug. 2011.
- [25] J. E. Schutt-Aine, "Latency insertion method (LIM) for the fast transient simulation of large networks," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 48, no. 1, pp. 81–89, Jan. 2001.
- [26] C. Wong, "EMTP modeling of IGBT dynamic performance for power dissipation estimation," *IEEE Trans. Ind. Appl.*, vol. 33, no. 1, pp. 64–71, Jan./Feb. 1997.
- [27] W. W. Hager, "Updating the inverse of a matrix," *SIAM Rev.*, vol. 31, no. 2, pp. 221–239, 1989.
- [28] B. Li, R. Yang, D. Xu, G. Wang, W. Wang, and D. Xu, "Analysis of the phase-shifted carrier modulation for modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 297–310, Jan. 2015.
- [29] D. Shu, V. Dinavahi, X. Xie, and Q. Jiang, "Shifted frequency modeling of hybrid modular multilevel converters for simulation of MTDC grid," *IEEE Trans. Power Del.*, vol. 33, no. 3, pp. 1288–1298, Jun. 2018.
- [30] 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.
- [31] Z. Huang, K. Sun, and Y. Mei, "A specific analysis model of three-level NPC inverter fed adjustable speed drive system with high accuracy," in *Proc. 17th IEEE Int. Conf. Elect. Mach. Syst.*, Hangzhou, China, 2014, pp. 914–919.
- [32] S. B. Monge, S. Somavilla, J. Bordonau, and D. Boroyevich, "Capacitor voltage balance for the neutral-point-clamped converter using the virtual space vector concept with optimized spectral performance," *IEEE Trans. Power Electron.*, vol. 22, no. 4, pp. 1128–1135, Jul. 2007.
- [33] *Saber User Guide, Version V-2004.06-SP1*, Synopsys, Inc., Mountain View, CA, USA, 2004.



**Zhen Huang** (S'18) received the B.Eng. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2012, and the M.Eng. degree in electrical engineering from Tsinghua University, Beijing, China, in 2015. He is currently working toward the Ph.D. degree at the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada.

He also has two-year work experience in power electronic industry. His research interests include high-performance simulation and control of power electronics.



**Venkata Dinavahi** (SM'08) received the B.Eng. degree in electrical engineering from the Visvesvaraya National Institute of Technology Nagpur, Nagpur, India, in 1993, the M.Tech. degree in electrical engineering from the Indian Institute of Technology Kanpur, Kanpur, India, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2000.

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.