Hardware-in-the-Loop Simulation of Power Electronic Systems Using Adaptive Discretization

M. Omar Faruque, *Member, IEEE*, and Venkata Dinavahi, *Senior Member, IEEE*

Abstract—This paper presents the hardware-in-the-loop (HIL) simulation of power electronic systems using a unique adaptive discretization technique based on the input of the system, where the coefficient matrices and system equations are changed while the simulation is running. The voltage-source-converter (VSC)-based HVDC system is used as a case study. Two z-transform-based discretization techniques, known as the step-invariant transformation (SIT) and the ramp-invariant transformation, and one of their derivative time-shifted SIT are applied for simulating the VSC-HVDC system. Discrete switching synchronization algorithms are used to accommodate the asynchronous events that take place in between two discrete simulation points. The HIL simulation is implemented using an off-the-shelf PC cluster interfaced with a digital controller. The simulator and the controller are connected through I/O ports, which facilitate the exchange of analog and digital signals. A 4-kW VSC-HVDC experimental setup is used to validate the simulation results, using the same digital controller as that in the HIL simulation to supply the necessary gate pulses for the experimental VSCs. A comparative study of the results obtained through offline, HIL simulation, and the experiment is presented.

Index Terms—Hardware-in-the-loop (HIL) simulation, HVDC transmission, power system transients, pulsewidth-modulated converters, real-time systems, z transforms.

I. INTRODUCTION

HARDWARE-IN-THE-LOOP (HIL) simulation has become a crucial stage in the design and development of digital controllers for a power electronic apparatus used at different power and voltage levels in a power system. Applications of this technology [1]–[7] include flexible ac transmission systems and HVDC at the transmission level, distributed generation and power quality regulation at the distribution level, and variable speed industrial drives at the utilization level. A comprehensive evaluation of all modes of operation of the controller under realistic system conditions is the primary requirement for the controller to be effective once it is installed in the field. HIL simulation offers the advantages of reduced

cost, fast repeatability of the tests, and the ability to subject the controller to risky contingency situations.

As the name implies, in HIL simulation, a part of the system is modeled and simulated in real time, while the remainder is the actual hardware, connected in closed loop through various I/O interfaces such as analog-to-digital (A/D) and digital-to-analog converters, and signal conditioning equipment. The simulation can be controlled by user-defined external inputs, e.g., closing and opening of switches to connect the components in the modeled system. HIL simulation can be of two kinds: controller HIL (C-HIL) simulation or signal HIL and power HIL (P-HIL) simulation. In C-HIL simulation, the power system, including the power electronic converters, is represented by a real-time model, while the digital controller is the device under test connected in closed loop with that model. This method is also known as rapid controller prototyping. There is no real power transfer in this method. The controller takes in the sampled voltages and currents from the simulator and provides control inputs in the form of switching signals. In contrast, in P-HIL simulation, a part of the power system is external to the simulator, thus requiring a transfer of real power to the external hardware.

In both types of HIL simulation, the use of an accurate discrete-time system model is paramount. Therefore, the method used to discretize the continuous-time system model is a key factor influencing simulation accuracy. Until now, most of the literature related to C-HIL simulation has focused on the accounting of discrete switching signals coming from the controller in a fixed-time-step simulation process [8]–[13]. However, the discretization methods used were the same as those used in offline Electromagnetic Transients Program simulation [14] such as the trapezoidal rule (TR). Although this method has been widely used, it can still cause errors in some cases due to spurious numerical oscillations [15]. In addition, at larger time steps, it can cause frequency warping and deviate from the actual response.

Various z-transform-based discretization techniques have been applied earlier for offline electromagnetic transient simulation [16], [17]. In those applications, a single discretization method is chosen for the entire simulation even though the performance of the discretization algorithm is found to vary with inputs. The accuracy of the simulation greatly depends on the selection of proper approximation (s to z domain) for a particular type of input to the system. In this paper, we propose an adaptive application of z-transform-based discretization methods such as step-invariant transformation (SIT), ramp-invariant

transformation (RIT), and time-shifted SIT (TSSIT), hitherto used for digital control and digital filter design [18] for HIL simulation. These methods are adaptive in the sense that they are excitation- or input-based, i.e., the discretization algorithm is selected based on the type of the system inputs. The methods are derived based on the assumption of input being held constant or being a ramp function for a time step, thereby producing error-free results if the supplied input to the system matches with the input assumed in the algorithm. During transients or disturbances, the excitation of the system changes abruptly. The proposed approach in such a situation is to switch to the proper discretization algorithm at the instant of change in the inputs. This technique represents the inputs correctly and minimizes the simulation errors, consequently improving the accuracy of the simulation due to the best representation of the discrete-time system. The challenge of using more than one discretization techniques in real-time simulation lies in the changes of coefficient matrices which contain time-consuming exponential terms. In an offline simulation, this is not a problem as the execution time is not a constraint; however, for real simulation, the simulation must be finished within the assigned time step. In real-time simulation, the coefficient matrices are precalculated for each discretization technique and switched when necessary.

This paper is organized as follows: In Section II, we present the fundamentals on the discrete-time simulation of a system using the z-transform-based techniques. In Section III, several examples are presented using simple RL and RLC circuits to compare the performance of these techniques vis-a-vis the TR and the exact solution. Section IV utilizes the adaptive discretization technique to perform a complete HIL simulation of a voltage source converter (VSC) HVDC system. Offline simulation and experimental implementation were done to verify the validity of the HIL simulation. In Section V, the three sets of results from the offline simulation, HIL simulation, and the experiment are analyzed and compared under both steady-state and transient scenarios. Conclusion is presented in Section VI.

II. z-Transform-Based Discretization Methods

Although several discretization techniques [14], [16], [17] are available in the literature, there is no unique, accurate, and stable equivalence between a continuous-time system and its discrete-time counterpart. The most commonly used discretization techniques for electromagnetic transient simulation are based on numerical integration algorithms. However, z-transform-based discretization methods can be used to produce more accurate, stable, and efficient results if they are selected and applied based on the type of input to the system.

z transform is a powerful operational method [18] in discrete-time analysis which considers only the sampled values of a function $y(t)$, i.e., $y(0), y(\Delta t), y(2\Delta t), \ldots$, where Δt is the sampling period. The z transform of a function $y(t)$ (where t is nonnegative) or of a sequence of values of $y(k\Delta t)$ (where k is zero or a positive integer) is defined as

$$Y(z) = \sum_{k=0}^{\infty} y(k\Delta t) z^{-k}$$
$$= y(0) + y(\Delta t) z^{-1} + y(2\Delta t) z^{-2}$$
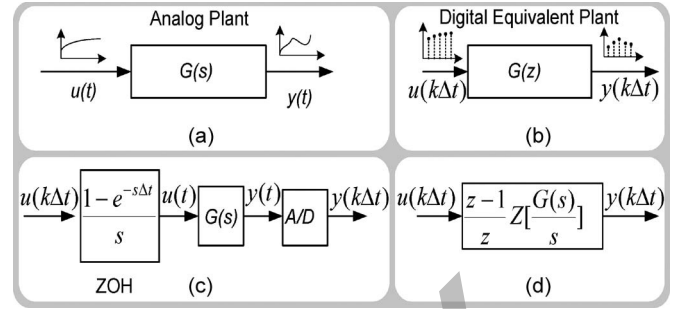$$+ \cdots + y(k\Delta t) z^{-k} + \cdots. \tag{1}$$

Fig. 1. (a) Analog plant. (b) Required digital plant for simulation. (c) Conversion to digital plant. (d) Converted digital plant.

Equation (1) implies that the z transform of any continuous-time function can be written in the form of a series, where z^{-k} indicates the position in time at which the amplitude of the function is $y(k\Delta t)$. The inverse z transform of this sequence of $y(k\Delta t)$ will give the values of $y(t)$ at the respective instants of time. Physical systems, which are linear and time invariant, can be represented as shown in Fig. 1(a) with an input $u(t)$, a transfer function $G(s)$, and an output $y(t)$. However, to perform computer simulation of this system, its digital equivalent shown in Fig. 1(b) is necessary, where the input is $u(k\Delta t)$, the transfer function is $G(z)$, and the output is $y(k\Delta t)$. If $y(t)$ is the response of a system with transfer function $G(s)$ and if its input is $u(t)$, digital simulation will yield responses $y(k\Delta t)$ only at times $t = 0, \Delta t, 2\Delta t, \ldots$. However, if the assumption of variations in inputs between any two consecutive discrete points does not match with the variation that is present in the real inputs, the response may not be accurate. If systems are discretized assuming that the input $u(t)$ varies either as a step function or a ramp function between two discrete instants, they are known as *SIT* and *RIT*, respectively.

Fig. 1(c) shows the conversion using SIT with the corresponding discrete-time transfer function $G(z)$ given as

$$G(z) = \frac{z-1}{z} \mathcal{Z} \left[\frac{G(s)}{s} \right] \tag{2}$$

which is equivalent to adding a zero-order hold at the beginning of the plant and an A/D converter after the plant. The conversion to $G(z)$ can also be done using RIT which follows

$$G(z) = \left[\frac{(1-z^{-1})^2}{\Delta t z^{-1}} \right] \mathcal{Z} \left[\frac{G(s)}{s^2} \right]. \tag{3}$$

These approximations of $G(s)$ yield exact responses only when the input is actually a step function or a ramp function, with changes occurring only at the sample instants. For other inputs, the approximation will introduce errors in the solution. These errors can be minimized through the careful selection of algorithms depending on the type of inputs. The response $y(t)$ of the system can be obtained as

$$y(t)|_{t=k\Delta t} = \mathcal{Z}^{-1} \left[G(z)U(z) \right] \tag{4}$$

where $U(z)$ is the z transform of $u(t)|_{t=k\Delta t}$.

If the s-domain transfer function is strictly proper, the SIT will introduce at least one pole in the origin, which creates a
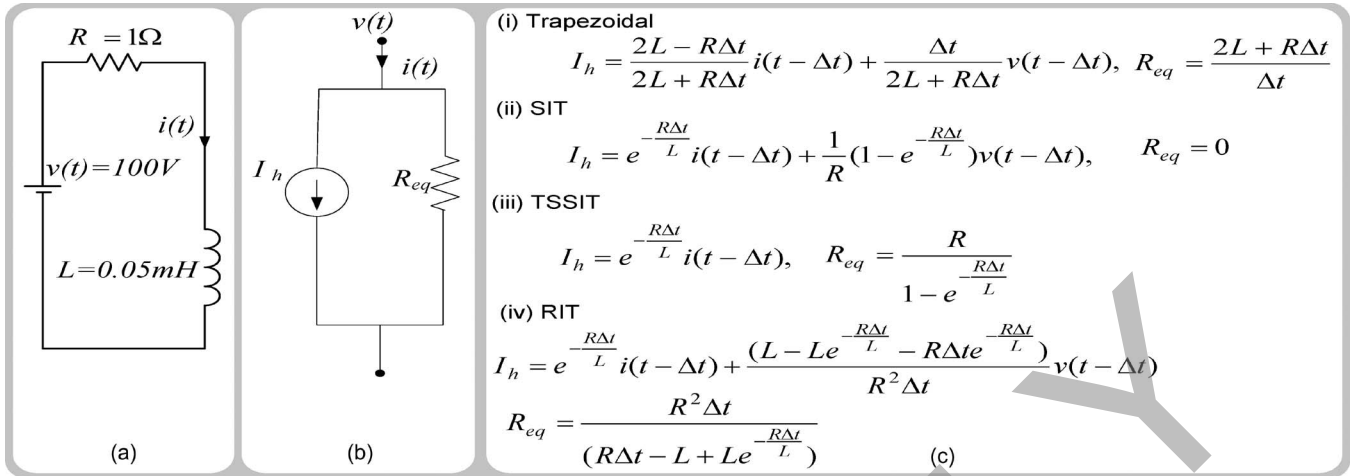
Fig. 2.   (a) RL network. (b) Norton equivalent. (c) Discrete-time equations using TR, SIT, TSSIT, and RIT.

pure time delay in the time-domain representation of the output. However, this delay can be compensated through adding zero(s) at the origin in the z-domain transfer function $G(z)$. In such a case, the response $y(t)$ will have a minimum phase delay, and the transformation will yield an *implicit* algorithm given as

$$y(t)|_{t=k\Delta t} = \mathcal{Z}^{-1} \left[z^r G(z) U(z) \right] \qquad (5)$$

where r is the number of added zeros. We call this transformation *TSSIT*.

A discrete-time system is considered stable if every bounded input sequence excites a bounded output sequence. The primary condition for a discrete-time system with a transfer function $G(z)$ to be stable is that every pole of $G(z)$ must lie inside the unit circle of the Z plane or have a magnitude of less than one. The relationship between s and z domain suggests that, if the system itself is stable in the s domain, it will remain stable in the z domain, when it is converted using either SIT or RIT. Details about system stability preservation while mapping from s to z domain are available in [18].

The transfer-function-based approach is commonly used to discretize the individual elements or branches, which are then used for a nodal-based solution technique. However, the complete system can also be solved using the state-space solution technique [19].

State-space equations for SIT, TSSIT, and RIT in the discrete-time domain are given in the Appendix.

III. MOTIVATIONAL EXAMPLES FOR THE PROPOSED DISCRETIZATION METHODS

Approximating a continuous transfer function into the discrete domain can be achieved by using several methods such as impulse invariant, step invariant, ramp invariant, pole–zero matching (root matching in [16]), bilinear, forward Euler (FE), and backward Euler. The main difference between these methods is due to the consideration of differences in the variation of inputs and the derivative of the state variables between two discrete points. If only the value of the input is considered at every discrete point, the transformation becomes impulse invariant. If the input is considered constant or linear over a time step,

the transformation is step invariant and ramp invariant, respectively. For root matching or exponential mapping, the pole and zeros are mapped into the z domain using the basic relationship $z = e^{s\Delta t}$. Then, the final value theorem is used to determine the value of the added constant for either step function or ramp function [16]. The SIT, TSSIT, and RIT methods do not use the final value theorem; instead, a step or ramp function is initially multiplied with the main transfer function [18]. Because of the similar type of input under consideration, the final form of some of the discretized equations obtained through TSSIT and RIT may be similar to the exponential equations given in [16]; however, there are differences. For example, for an RL branch, the root-matching (exponential) equation matches with TSSIT, but it does not match with the RIT.

To compare the accuracy of SIT, TSSIT, and RIT with other low-order discretization techniques such as the FE and the TR, simple circuits such as RL and RLC are simulated using all of these methods. The circuits are excited by step functions and sinusoidal voltage sources. Both nodal and state-space approaches have been adopted to show that the algorithms perform with acceptable accuracy using both approaches.

Example I—RL Circuit Solved Using the Nodal Approach

Fig. 2(a) shows an RL circuit that has a time constant $\tau = 50~\mu s$ and is excited by a step function 100 V. The Norton equivalent of the RL circuit is shown in Fig. 2(b), and Fig. 2(c) shows the discrete-time equations obtained using the TR, the SIT, the TSSIT, and the RIT.

To demonstrate the differences in accuracy, the network is simulated by using the FE, the TR, the SIT, the TSSIT, the RIT, and the exact solution (obtained from Laplace solution). A time step of $\Delta t = \tau = 50~\mu s$ has been chosen for the simulation of the discretization algorithm; however, an extra set of results is also plotted for the exact solution and TSSIT using a 1-μs time step. The responses produced by different methods are shown in Fig. 3(a). It is clear that the simulation results obtained by the TSSIT match with the exact solution, whereas other methods show errors. It was found that, irrespective of the time step used for simulation, the exact solution always matches the TSSIT response for a step-function input. The response produced by
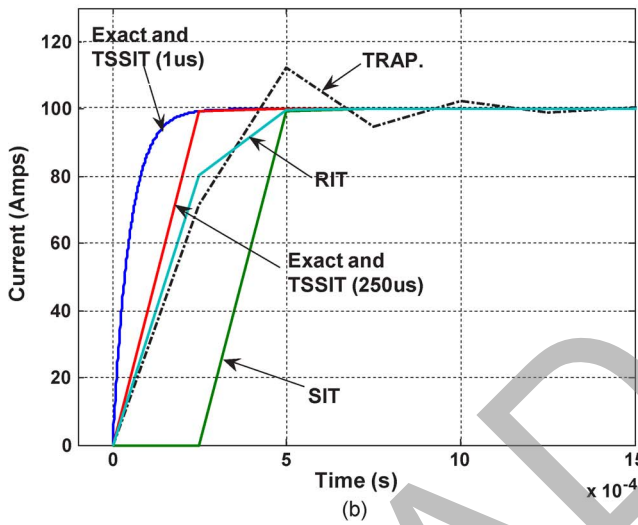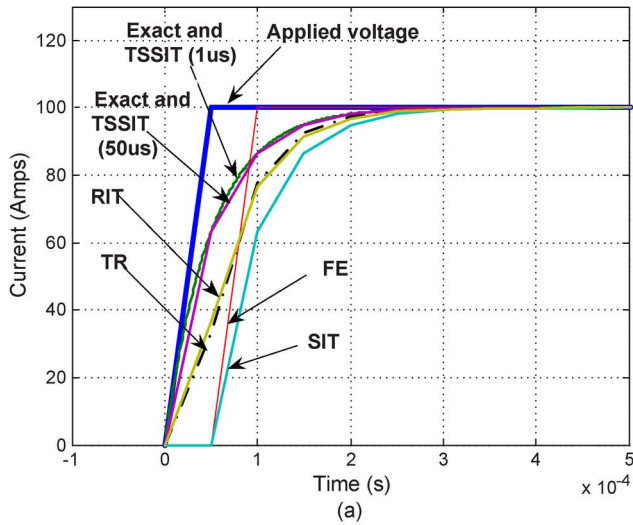
Fig. 3. Step response (a) for $\Delta t = \tau = 50 \ \mu s$ and (b) for $\Delta t = 5\tau = 250 \ \mu s$.

Fig. 4. Network response at a time step of $\Delta t = 10 \ \mu s$: (a) For a sinusoidal input. (b) Zoomed view.

SIT is similar to the TSSIT response except that SIT creates one time-step delay in the output. As shown in Fig. 3(b), for a time step of $\Delta t = 5\tau = 250 \ \mu s$, FE does not converge, and the TR shows oscillations even though SIT, TSSIT, and RIT are stable. The TSSIT response again matches with the exact response at both 1 and 250 μs. The oscillation in the TR decays slowly and finally settles to a steady-state value. However, the RIT produces more accurate and stable response than the trapezoidal response, and it settles to a steady-state value faster than TR without showing any oscillation. This concludes that the proposed methods (SIT, TSSIT, and RIT) can be applied even at larger time steps when the TR fails to response accurately.

The same RL circuit is now excited by a sinusoidal voltage with a frequency of 2 kHz. A time step of $\Delta t = 10 \ \mu s$ was used to simulate the network using all the methods. The responses obtained through all these methods are shown in Fig. 4. As can be seen from Fig. 4, the steady-state response of the network obtained through TR and RIT closely matches with the exact response. The other methods show larger errors. However, if the input sinusoidal voltage is changed suddenly, the TSSIT
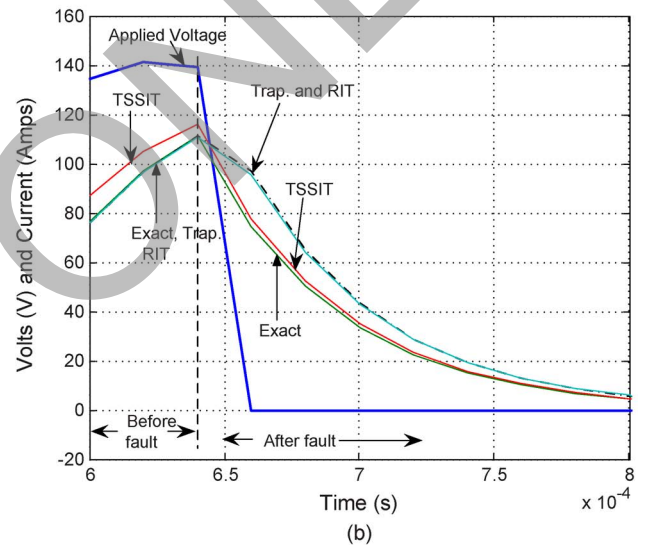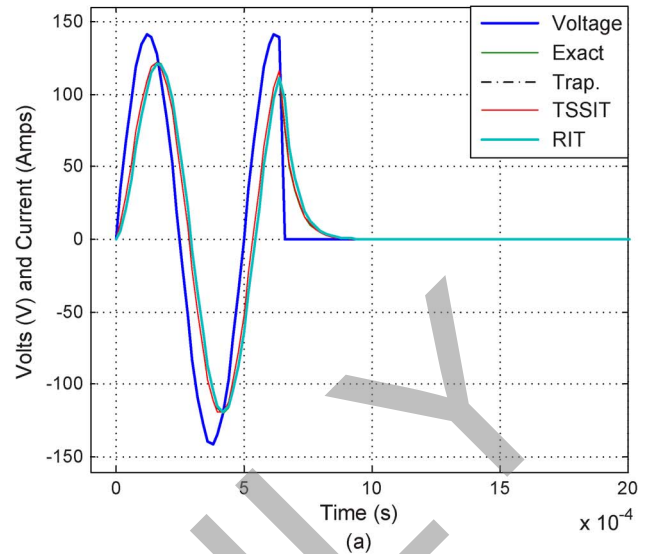
response becomes closer to the exact response, as shown in the zoomed view in Fig. 4(b). It can be concluded that, for a system with sinusoidal input, RIT performs closer to the exact solution and that, if there is any abrupt input change such as during a fault, TSSIT would outperform the others.

Example II—RLC Circuit Solved Using the State-Space Approach

The use of SIT and RIT in the state-space solution technique has long been used in solving electronic circuits [19]. An example case has been described here to demonstrate the suitability and comparative advantage of SIT and RIT over TR for solving an RLC circuit. A series RLC circuit, shown in Fig. 5, was solved using the state-space solution technique. The input to the system is a step function given as

$$U(t) = 10u(t). \tag{6}$$

The parameters of the RLC circuit are chosen arbitrarily, keeping in mind that the step response becomes oscillatory with
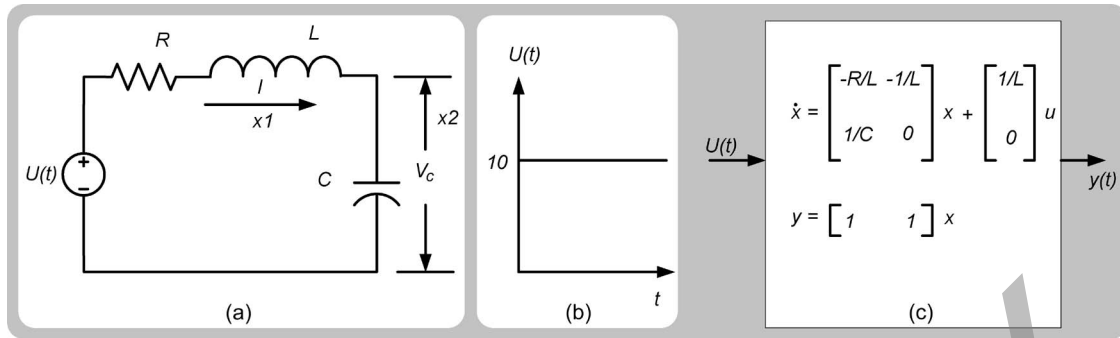
Fig. 5. (a) RLC network. (b) Step-function input. (c) Equivalent state-space model.
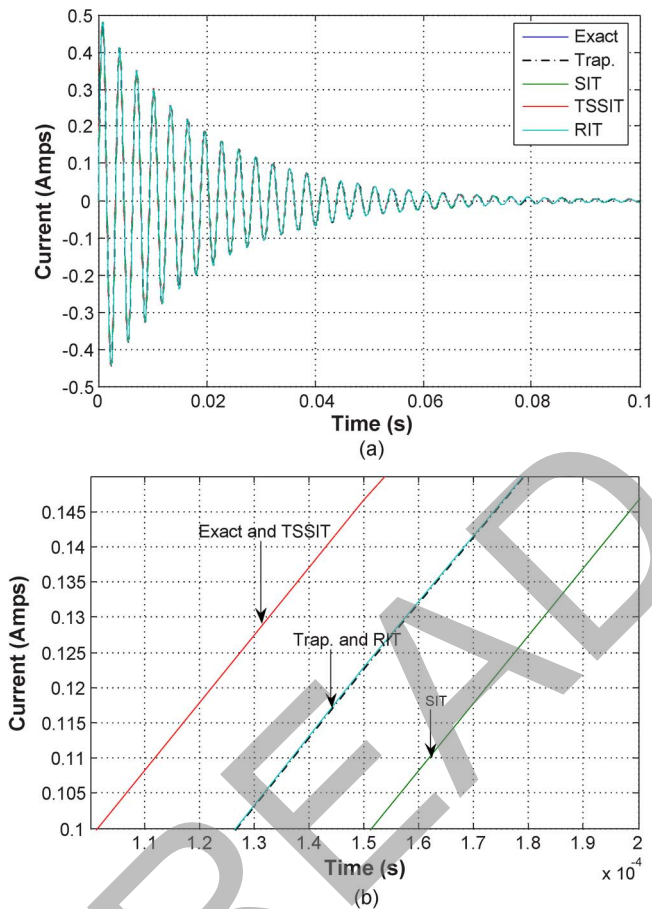

Fig. 6. (a) Step response of an RLC network solved by the state-space solution technique with different discretization methods at a time step of $\Delta t = 50 \ \mu$s. (b) Zoomed view of the response.

a decaying amplitude. A value of $R = 1 \ \Omega$, $L = 10$ mH, and $C = 25 \ \mu$F produces the following state-space equation:

$$\dot{x}(t) = \begin{bmatrix} -100 & -100 \\ 40\,000 & 0 \end{bmatrix} x + \begin{bmatrix} 100 \\ 0 \end{bmatrix} u \qquad (7)$$

$$y = \begin{bmatrix} 1 & 1 \end{bmatrix} x. \qquad (8)$$

The aforementioned state-space equation can be simulated using the discretization algorithms such as TR, SIT, TSSIT, and RIT. The simulation results obtained using a time step of 50 μs are shown in Fig. 6. It can be seen that the solution obtained through the TSSIT method matches accurately with
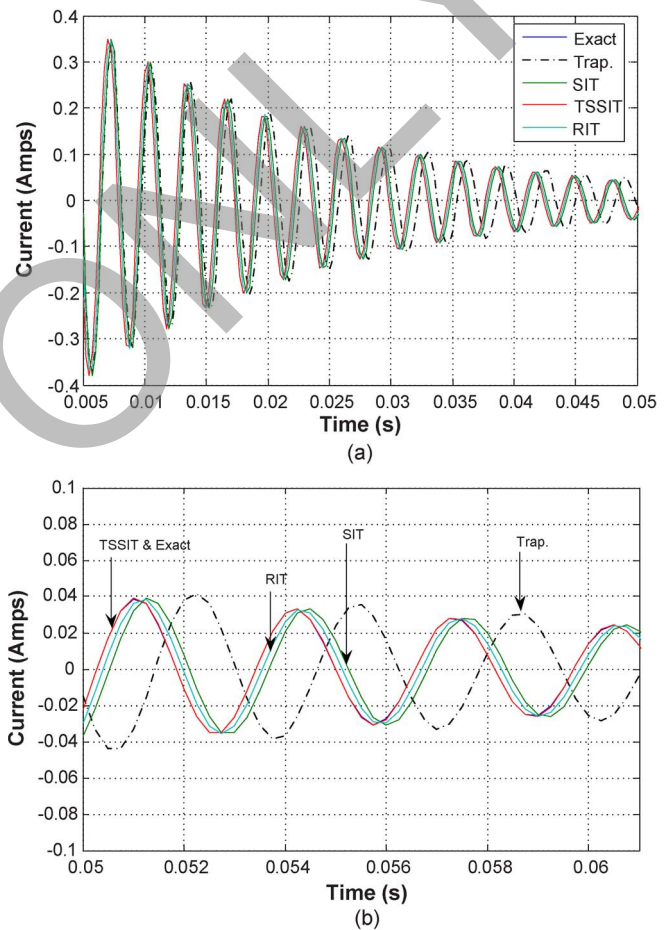
Fig. 7. (a) Step response of an RLC network solved by the state-space solution technique using different discretization methods with a time step of $\Delta t = 250 \ \mu$s. (b) Zoomed view of the response.

the exact solution. However, the SIT shows a phase error due to its inherent delay of one time step. The RIT has been found to match closely with the trapezoidal method, which was the case in the nodal approach. However, when the same circuit is simulated using a time step of $\Delta t = 250 \ \mu$s, it was found that the TSSIT method still matches with the exact method, while RIT and SIT produce errors (Fig. 7). RIT produces less error than SIT as SIT suffers from an inherent time-step delay. On the other hand, TR shows a frequency error which is not seen in any other case. This phenomenon is known as frequency warping, which normally occurs when the TR is used with a large time step. Case studies show that the type of input plays an important
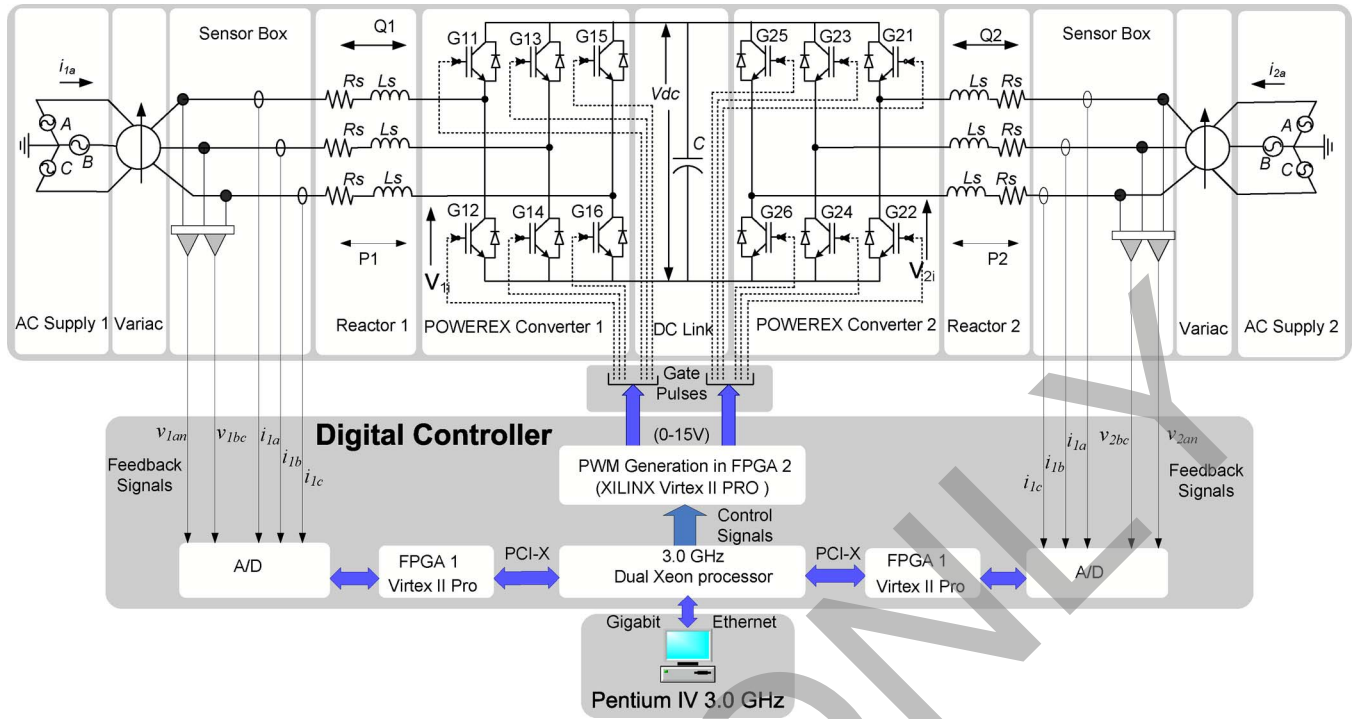
Fig. 8. Schematic of the VSC-HVDC system used for the HIL simulation and the experimental setup.

role in the accuracy of the simulation results using z-transform-based discretization methods. z transform alone does not consider how the input is varied between discrete points; however, algorithms developed based on the variation of inputs between discrete points such as SIT, TSSIT, and RIT produce more accurate results if the change in input is taken into account properly.

IV. HIL SIMULATION CASE STUDY: VSC-HVDC SYSTEM

VSC-HVDC is a VSC-based dc transmission technology, where force-commutated switching devices such as gate turn-off thyristors or insulated-gate bipolar transistors (IGBTs) are operated through a pulsewidth modulation (PWM) scheme. With this technology, it is possible to generate voltages at any magnitude, angle, and frequency (up to a certain limit) through changes in the control and PWM scheme. Because of the controllability of the VSCs, active power can be transferred from one side to another, and reactive power can be generated independently.

Fig. 8 shows the schematic diagram of the VSC-HVDC test system and its controller used for the HIL simulation and the experiment. It mainly consists of two ac sources, phase reactors on both sides, two converters, and a dc link. The two ac supplies of the network are represented by three-phase Y-connected and grounded infinite sources with no source impedance. The two converters, with one operating as an inverter and the other as a rectifier, are built of IGBTs and connected back-to-back through a dc-link capacitor.

A. Operation of the VSC-HVDC System

The converters used in a VSC-HVDC system can be considered as variable voltage sources as they are capable of generating voltages of any magnitude, phase angle, and frequency. With respect to the dc-link voltage V_{dc}, the instantaneous voltage at the terminal of converter 2 V_{2i} can be given as

$$V_{2i} = K m_a V_{dc} \sin(\omega t + \delta) \tag{9}$$

where K is a constant whose value depends on the modulation scheme, m_a is the modulation index defined by the ratio of the peak value of the modulating wave to the peak value of the carrier wave, ω is the fundamental frequency, and δ is the phase shift with respect to the ac terminal voltage. The phase-shift angle δ and the modulation index m_a can be controlled to generate any combination of voltage magnitude and phase angle (limited by the dc voltage).

B. Control of the VSC-HVDC System

The most important aspect of a VSC-HVDC is its ability to control active and reactive power independently. The active power is controlled by controlling the dc-link voltage, and the reactive power is controlled by changing the ac voltages at the converter terminals. Using the $abc-\alpha\beta-dq$ transformations and a vector control technique [22], the control schemes for both converters were developed. The control schemes for the two converters are similar with small differences in the derivation of the reference signals. The decoupled control scheme for VSC 1 is given in Fig. 9. The objective of the control scheme is to determine the required m_a and δ for any particular level of power transfer in such a way that any change in the reference power level will be reflected through a change in these two parameters. This will ultimately change the gating signals, and power transfer will be maintained dynamically without any interruption.
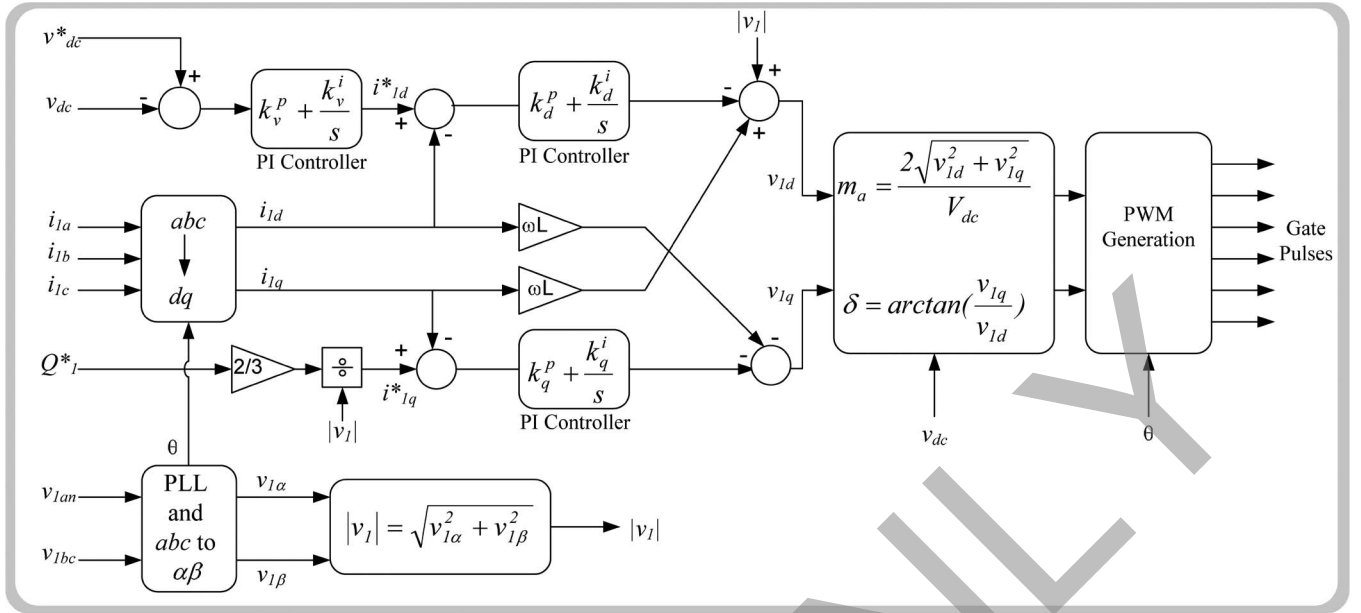
Fig. 9.   Decoupled vector control scheme for VSC 1.

C. VSC-HVDC System Model

1) AC Systems: The ac systems on both sides of the HVDC network are modeled as pure sinusoidal voltages. A 2.5-mH inductance with a resistance of 0.2 Ω is modeled as series impedances in all three phases. The system elements were discretized using RIT as it produces the minimum error for sinusoidal inputs in a steady-state situation; however, for transients that cause abrupt input changes, TSSIT was used. The current at VSC 1 is thus given by

$$i_{1n}(t) = \Gamma\left[i_{1n}(t - \Delta t)\right] + \Lambda\left[v_{1n}(t) - v_{1ni}(t)\right]$$
$$+ \Psi\left[v_{1n}(t - \Delta t) - v_{1ni}(t - \Delta t)\right] \quad (10)$$

where $n = \{a, b, c\}$, $\Gamma = e^{-(R\Delta t/L)}$, $\Lambda = (R\Delta t - L + Le^{-(R\Delta t/L)})/(R^2\Delta t)$, and $\Psi = (L - Le^{-(R\Delta t/L)} - R\Delta te^{-(R\Delta t/L)})/(R^2\Delta t)$. In these equations, Γ, Λ, and Ψ are constants whose values depend on the system parameters and the time step Δt. These constants remain the same unless the system parameters or the time step is changed and are calculated once at the beginning of the simulation. The same equations are used for modeling the ac system of VSC 2.

2) VSCs: The switching function model used in [9] was chosen to model the VSCs.

To get the instantaneous voltages showing high-frequency switchings, voltages can be expressed in terms of the discrete switching functions S_k, $k = \{1, 3, 5\}$, which control the upper switches in each leg of the converter. When $S_k = 1$, the switch is on, and when $S_k = 0$, the switch is off. The lower switches in each leg of the converter are switched in a complementary manner with an added dead time. The output voltages of the converter with respect to the negative dc bus N are given as

$$v_{kN} = S_k V_{\text{dc}}, \qquad k = \{1, 3, 5\}. \quad (11)$$

Under balanced conditions, the converter output voltages with respect to the ac system neutral n are given as

$$v_{kn} = \frac{2}{3}v_{kN} - \frac{1}{3}\sum_{\substack{i=\{a,b,c\}\\i\neq k}} v_{iN}. \quad (12)$$

3) DC Link: The voltage in the dc link is maintained by controller 1 which uses PWM to keep the dc-link voltage constant. The voltage and currents in the dc link are given as

$$i_{\text{dc1}}(t) = S_{11}(t)i_{1a}(t) + S_{13}(t)i_{1b}(t) + S_{15}(t)i_{1c}(t) \quad (13)$$

$$i_{\text{dc2}}(t) = S_{21}(t)i_{2a}(t) + S_{23}(t)i_{2b}(t) + S_{25}(t)i_{2c}(t) \quad (14)$$

$$V_{\text{dc}}(t) = I[i_{\text{dc1}}(t) + i_{\text{dc1}}(t - \Delta t) + i_{\text{dc2}}(t) + i_{\text{dc2}}(t - \Delta t)]$$
$$+ J[V_{\text{dc}}(t - \Delta t)] \quad (15)$$

where S_{11}, S_{13}, and S_{15} are the states (one or zero) of the top IGBTs of VSC 1 and S_{21}, S_{23}, and S_{25} are the states of the top IGBTs of VSC 2. I and J are again two constants whose values depend on the parameters, the time step, and also on the method of discretization used for simulation.

D. Offline Simulation, HIL Simulation, and Experimental Validation

1) Offline Simulation: An offline simulation of the VSC-HVDC system and its controllers is performed using the C language. The objective is to ensure that the modeling, control, and solution techniques described in the previous section are accurate and efficient and would pose no difficulty in real-time implementation. The simulation is carried out using Microsoft Visual C/C++ v.6.0. A 2-kHz switching frequency is used for the carrier, and a 20-μs simulation time step is used for the offline simulation. The results from the offline simulation have also been verified using the PSCAD/EMTDC and MATLAB/Simulink software.
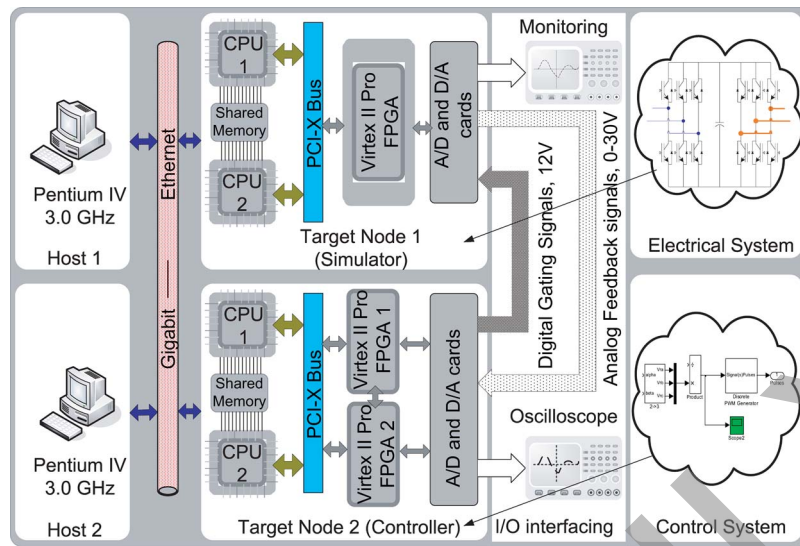
Fig. 10.   Block layout of the hardware setup for the HIL simulation of the VSC-HVDC system.

2) HIL Simulation: The study of the HIL simulation of the VSC-HVDC was conducted using a PC-cluster-based real-time simulator described in [20]. The hardware architecture for the HIL simulation is shown in Fig. 10. *Target* 1 is used as the simulator, while *Target* 2 is used as the controller. *Host* 1 is used to prepare the model for simulation using *Target* 1. It can also be used to monitor the real-time simulation results collected from *Target* 1. Similarly, *Host* 2 is used for developing the control scheme, and once the controller is ready, it is loaded on *Target* 2 for implementation. Thus, *Target* 2 works as the digital controller whose inputs are analog feedback signals (± 15 V) and the outputs are gate signals (0–15 V) that are sent to the simulator. The HIL simulation and the controller are independently controlled by both *Hosts* 1 and 2.

The simulator node *Target* 1 has one field-programmable gate array (FPGA), while the controller node *Target* 2 has two FPGAs. The additional FPGA (FPGA 2) in *Target* 2 is dedicated for generating gating signals by comparing the carrier signals (generated inside the FPGA) with the reference signals (produced by the control algorithm in the Xeon CPU). The FPGAs are Xilinx Virtex-II Pro, with 11 088 logic cells and an IBM power processor inside, and operate at a frequency of 100 MHz. This gives a 10-ns resolution, which is the time step used for generating a triangular carrier signal for PWM. On the one side of *Target* 2, FPGA 1 is connected to the PCI-X slot of the computer, and on the other side, it is connected to the FPGA 2 board and the I/O carriers.

Simulink, RT-LAB, SimPowerSystems, and C program are the main tools used for developing, loading, and running the models for HIL simulation. For modeling the electrical system of the VSC-HVDC, S-function is used to wrap the C program in the Simulink environment. Once the models were prepared, they were loaded on the two *target* nodes using RT-LAB [21]. As shown in Fig. 11, the two *targets* are physically connected through I/O channels using standard wires. The I/O channels are used to transmit analog and digital signals between *targets*. Six 0–15-V gating signals come out through the digital outputs of *Target* 2 and are fed to the simulator through the digital
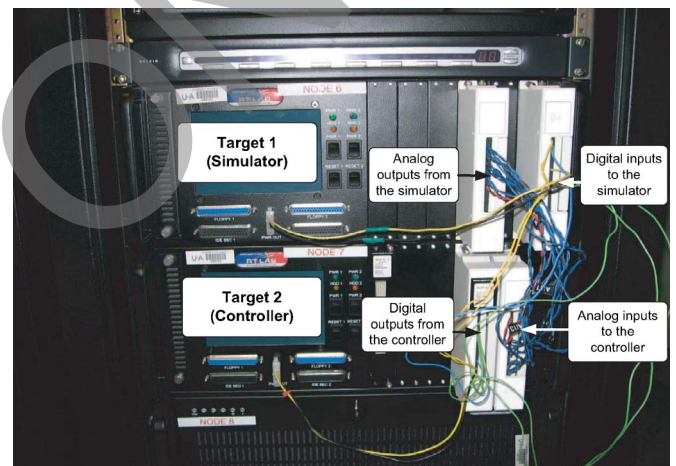

Fig. 11.   Snapshot of the hardware setup for HIL simulation of the VSC-HVDC system shows I/O connections.

inputs of *Target* 1. Switching events are precisely captured using FPGA 1. During each time step, the digital input lines are checked continuously for transitions. The hardware reports the moment of a *state* change relative to the start of the computation step. Rising, falling, or both transitions can be reported as requested by the user. During one calculation step, all captured events are stored on the FPGA-based event detector card, and they are uploaded by the block at the beginning of the next calculation step [21].

3) Experimental Setup: In order to validate the HIL simulation results, a 4-kW VSC-HVDC system was built in the laboratory. Two 2.5-mH 600-V 18-A inductors were used on the two ac sides. Two modules of IGBT-based three-phase converters were connected to construct the VSC-HVDC circuit. Each converter consists of six IGBTs, a gate drive board with built-in protection, LEM sensors to measure currents, and a large heat sink with a blower for cooling the IGBTs. The gate drive board is mounted on top of the module, and it implements a default 2-μs dead time to prevent dc-link short circuit. Similar to the HIL setup, the *Target* 2 node in the PC cluster is used
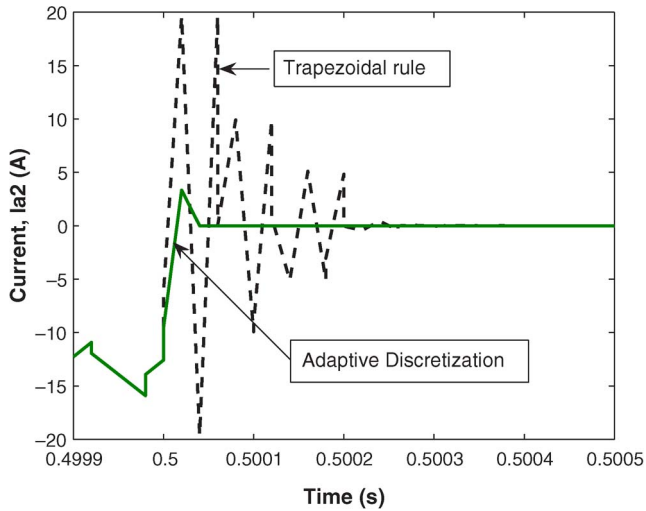
Fig. 12. Responses at converter 2 current obtained by TR and the adaptive algorithm when the supply voltage was interrupted.

as the digital controller, where the same control scheme is implemented. The experiment was monitored and controlled from a *Host* 2 computer.

V. RESULTS AND DISCUSSION

Before showing the detailed results, the increased accuracy of adaptive discretization over the TR is demonstrated in Fig. 12 using an offline simulation. A loss in the three-phase power supply at converter 2 was simulated. In this particular case study, the following steps were used to implement the adaptive discretization procedure.

Step 1) System matrices obtained through TSSIT were used for the initial time steps during the initialization and then switched to those from RIT until steady state was reached.

Step 2) A supply interruption at converter 2 is planned at time t when TSSIT is used instead of RIT.

Step 3) The simulation returned to using the system matrices obtained through RIT after the supply voltage was restored.

As shown in Fig. 12, the TR shows spurious oscillations when the current at converter 2 goes to zero due to the loss of supply. Experimental results verified the fact that there was no oscillation in the current at such instants.

A. Steady State

For the steady-state results, the initial start-up transients are excluded. Two cases have been considered for all three categories, and closed-loop results are given.

Case 1—Positive Power References: In this case, all power reference quantities are set to positive values as given in the following:

1) $V_{dc} = 500$ V and $P_2 = 3000$ W;
2) $Q_1 = 2000$ VAr and $Q_2 = 2000$ VAr.

Figs. 13 and 14 show the real-time oscilloscope traces obtained from the HIL simulation and the experiment, which are found to be in close agreement. A negligible mismatch between
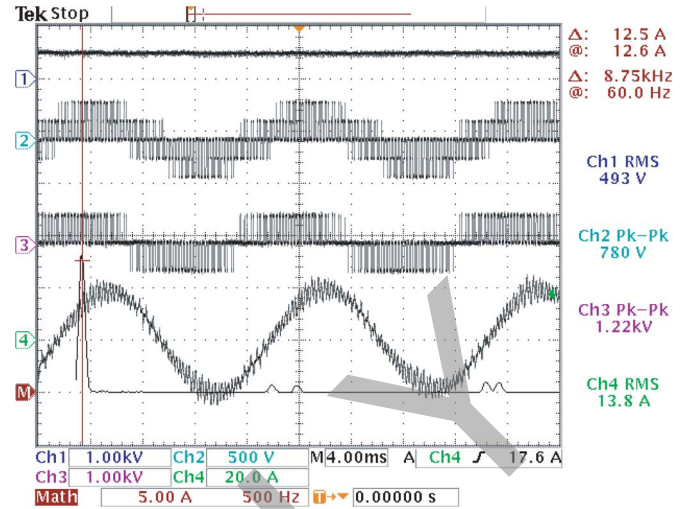

Fig. 13. Oscilloscope trace of steady-state dc-link voltage V_{dc}, phase voltage V_{a2}, line-to-line voltage V_{ab2}, and phase-a current I_{a2} and FFT of the current I_{a2} for HIL simulation (Ch1: 1 kV/div, Ch2: 500 V/div, Ch3: 1 kV/div, Ch4: 20 A/div).
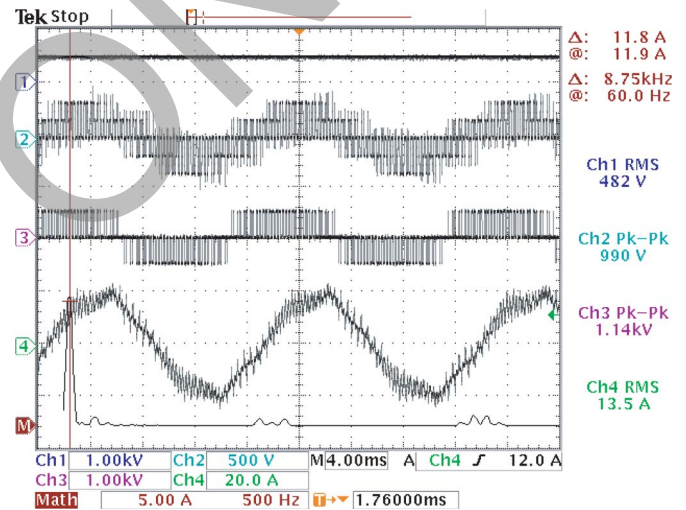

Fig. 14. Oscilloscope trace of steady-state dc-link voltage V_{dc}, phase voltage V_{a2}, line-to-line voltage V_{ab2}, and phase-a current I_{a2} and FFT of the current I_{a2} obtained from the experiment (Ch1: 1 kV/div, Ch2: 500 V/div, Ch3: 1 kV/div, Ch4: 20 A/div).

dc voltages is observed, which is mainly due to the voltage probes and I/Os. In the experimental case, the dc voltage was measured using a dc voltmeter and was found in the range of 497–502 V, even though the oscilloscope trace shows it slightly lower (482 V). The phase voltage and the line-to-line voltage for all three cases are similar and as expected. Neglecting small distortion which is due to the distorted supply voltage, unbalance, and differences in parameters between the simulation and the experiment, the current waveforms look similar. The frequency of the currents is found to be 60 Hz for both the HIL simulation and the experimental results. The rms values are found to be 13.8 A for the HIL simulation and 13.5 A for the experimental results.

B. Transients

Case 2—DC Voltage Reference Is Changed While Power References Remain Unchanged: In this case, the power
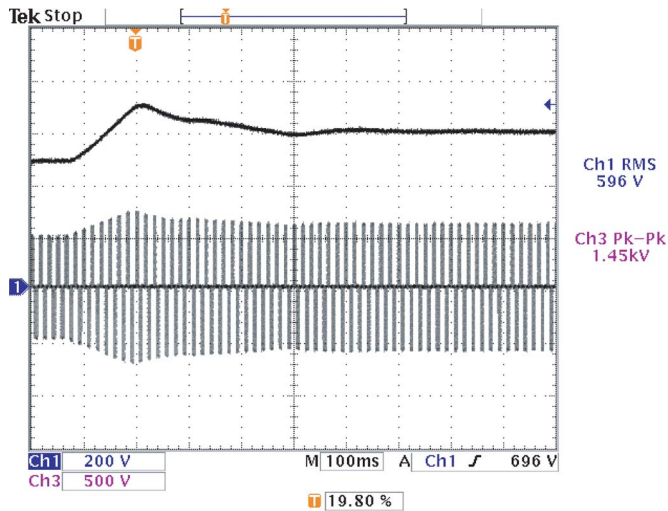
Fig. 15. Oscilloscope trace of transient responses in the dc-link voltage $V_{\rm dc}$ and the line-to-line voltage v_{ab2} for HIL simulation (Ch1: 200 V/div, Ch2: 500 V/div).
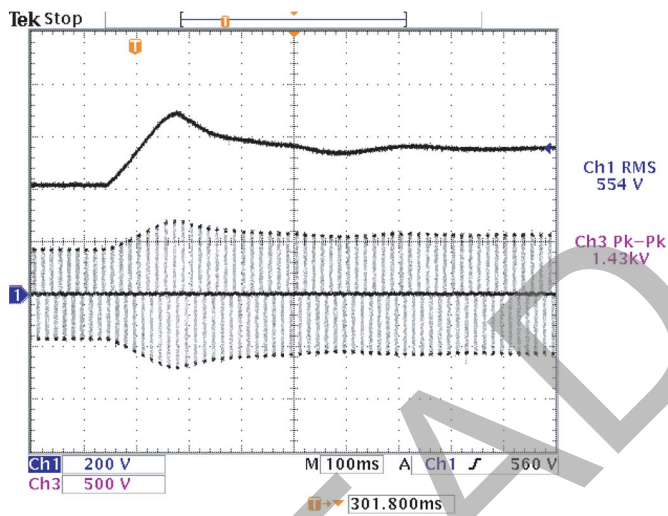

Fig. 16. Oscilloscope trace of transient responses in the dc-link voltage $V_{\rm dc}$ and the line-to-line voltage v_{ab2} obtained from the experiment (Ch1: 200 V/div, Ch2-500 V/div).

references are kept at $P_2 = 3000$ W, $Q_1 = 2000$ VAr, and $Q_2 = 2000$ VAr, while the dc-link voltage reference is changed. When the system starts to run under closed loop, the initial dc voltage reference is kept at 420 V. For offline simulation at $t = 0.4$ s and for HIL and the experiment at $t = 50$ s, the reference voltage is changed to 560 V in the first step and then to 440 V in the second step.

The transitions in the dc-link voltage and the line-to-line voltage at the terminal of VSC 2 for both HIL simulation and the experiment are captured on the oscilloscope and shown in Figs. 15 and 16, respectively. It can be observed that, for both HIL simulation and the experiment, the responses are very similar. From Fig. 17, it can be seen that the dc-link voltage has settled to its new values after the initial transients. There are slight differences between the transient peaks and their settling times, which can be attributed to the following reasons.

1) The changes in references were performed at different times. Because of the large amount of data generated, the

offline simulation was performed for only 1 s. For the HIL simulation and the experiment, it takes a while (approximately 20 s) to bring the system into operation because of the involvement of manual operations of switchgears. Therefore, the time of changing the step in the dc voltage for all three cases could not be maintained at the same instant.

2) The proportional-integral parameters found from the control design were used in the offline simulation. These gains needed slight retuning ($\pm20\%$) when used in HIL simulation and the experiment due to the differences in the system parameters between the real and measured values.

The change in dc-link voltage more or less affects all the power levels. Momentary disturbances are created in all power levels which die out very fast. Other than some minor discrepancies in the transient magnitude and response, the three cases produce stable results which are in close agreement.

Case 3—Power References Are Changed While DC-Link Voltage Left Unchanged: Here, all the three power references went through multiple changes, including power reversal, while the dc-link voltage is kept unchanged. The responses are shown in Fig. 18, where it was found that the dc-link voltage remained almost unaffected even though power demands went through multiple changes. Small spikes were observed in the experimental results and HIL simulation which are due to the changes of various power references. However, offline simulation failed to exhibit any spike or disturbances in the dc-link voltage. It can be observed that, in the case of HIL simulation and the experiment, while a change takes place in any of the power references, slight disturbances are observed in the rest of the power levels; however, offline simulation seems immune to those circumstances.

From all these case studies, it can be concluded that HIL simulation is more realistic and in close agreement to the experimental results than the offline simulation. In most cases, HIL simulation showed results that are similar to the experimental results which validate the accuracy of the HIL simulation obtained at a time step of 10 μs. However, minor discrepancies are observed due to the following differences or limitations.

1) In the offline simulation, the source voltage is considered perfectly sinusoidal and balanced, whereas in the HIL simulation and experiment, the supply voltage was unbalanced and distorted.

2) The amplitude of the supply voltage in the offline simulation and HIL simulation is assumed constant. However, in the case of the experiment, slight variation is observed between the inductive mode of operation and the capacitive mode of operation of the VSCs.

3) Differential voltage probes and current sensors contribute errors in the measurements due to the delay and inherent dc offsets.

4) Minor differences exist between the values of the system parameters used in the HIL simulation and in the experiment. The effect of parameter variation with temperature was not considered in the HIL simulation.
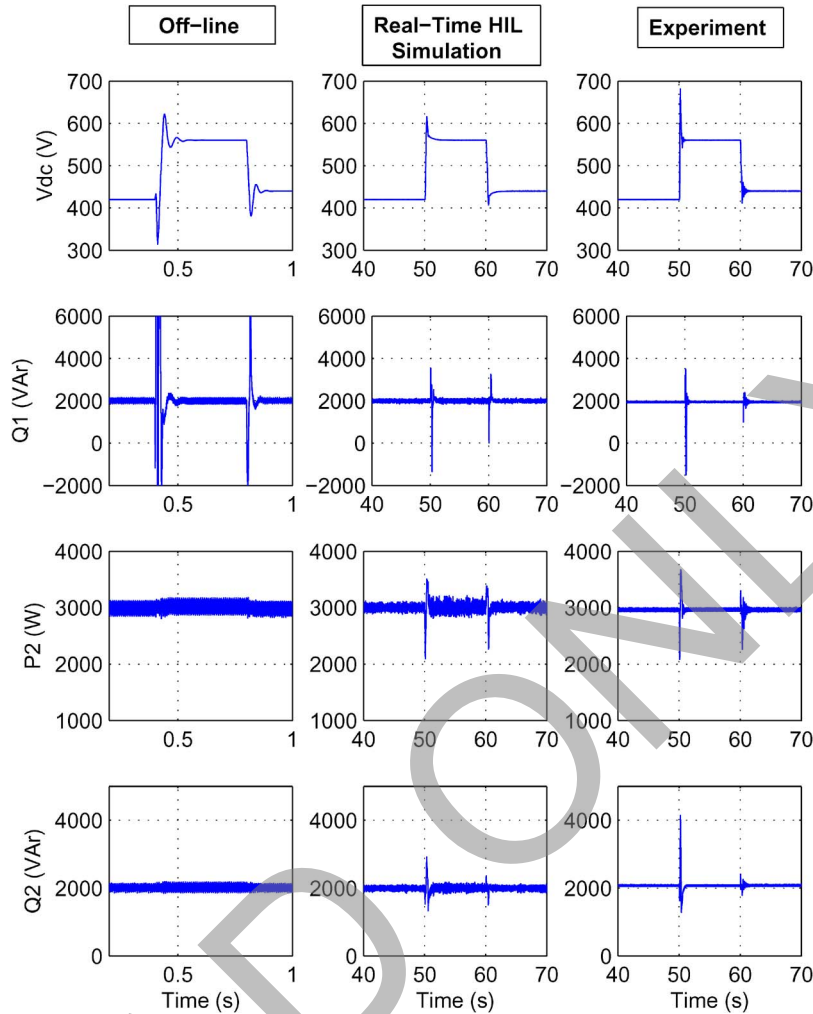
Fig. 17.   Offline simulation, HIL simulation, and experimental results of dc-link voltage V_{dc}, VSC 1 reactive power Q_1, and VSC 2 power P_2 and Q_2 for changes in the dc-link voltage reference.

VI. CONCLUSION

This paper has proposed z-transform-based discretization methods for HIL simulation of power electronic systems. The comparative study with the existing numerical integration techniques has revealed that the proposed methods are more accurate and stable if they are selectively applied based on the nature of the excitation of the system. In the light of this investigation, the following conclusions can be drawn.

1) The SIT is an explicit discretizing method which introduces phase error in the simulation due to its inherent delay. However, the advantage of this method is that it is usable in real-time simulation due to its explicit characteristics.
2) The TSSIT is an implicit method and thus has no delay in the response. It produces accurate results in case of transients created by sudden change of the inputs.
3) The RIT and the TR give similar performance at small time steps; however, with increasing time step, ramp invariant is found to be more stable and accurate than the TR.
4) Unlike the TR, the use of the proposed adaptive discretization technique does not require corrective measure about numerical oscillation or frequency warping.

A PC-cluster-based real-time simulator was used to perform the HIL simulation and the implementation of the controller in the experiment. For all three cases, namely, offline simulation, HIL simulation, and the experiment, closed-loop cases were studied, where the system went through steady-state and transient situations at various power levels. Close agreement between all these sets of results was observed. In particular, the HIL simulation showed greater accuracy by being closest to the experimental results.

APPENDIX

The digital equivalent transfer function $G(z)$ of an analog plant $G(s)$ can also be obtained from the state-variable equations

$$\dot{\mathbf{x}}(t) = \mathbf{A_c}\mathbf{x}(t) + \mathbf{B_c}\mathbf{u}(t) \tag{16}$$

$$\mathbf{y}(t) = \mathbf{C_c}\mathbf{x}(t) + \mathbf{D_c}\mathbf{u}(t) \tag{17}$$

which are the realization of $G(s)$. If the input $\mathbf{u}(t)$ is step invariant

$$\mathbf{u}(t) = \mathbf{u}(k\Delta t), \qquad k\Delta t \le t < (k+1)\Delta t;$$
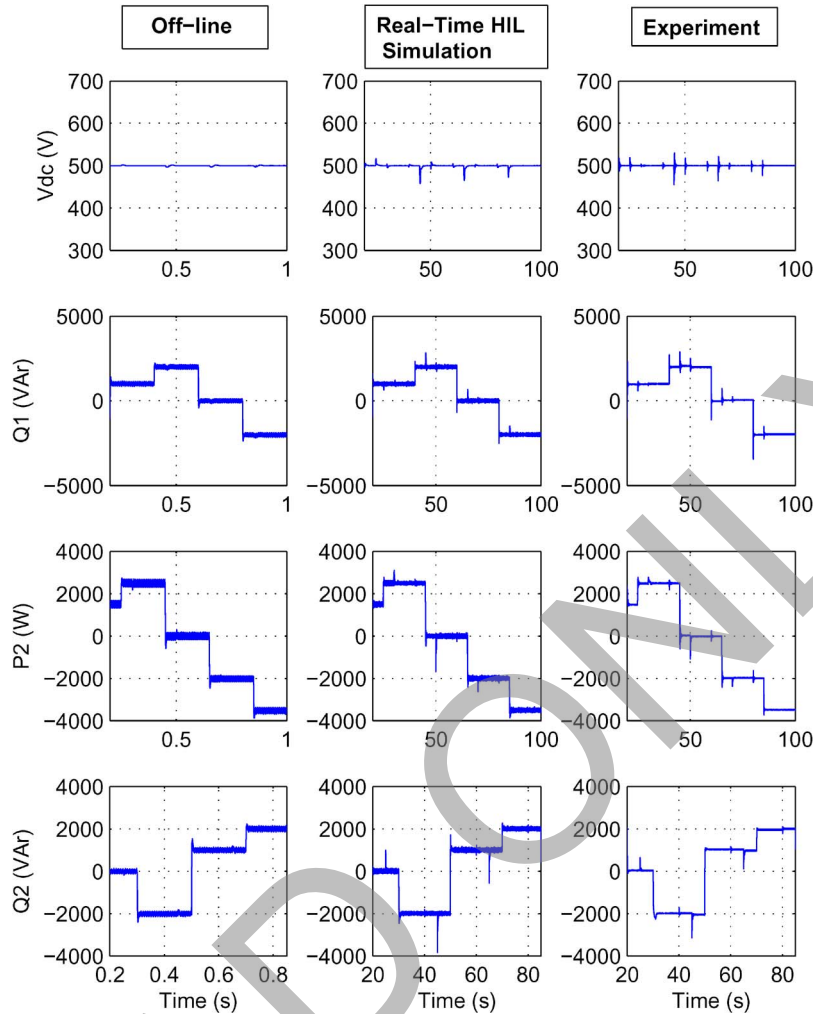$$k = 0, 1, 2, 3, \ldots \tag{18}$$

Fig. 18.   Offline simulation, HIL simulation, and experimental result of dc-link voltage $V_{\rm dc}$, VSC 1 reactive power Q_1, and VSC 2 power P_2 and Q_2 when power references were changed.

then the output $y(t)$ at $t = k\Delta t$ can be described as

$$\mathbf{x}(k+1)\Delta t = \mathbf{A_d}\mathbf{x}(k\Delta t) + \mathbf{B_d}u(k\Delta t) \qquad (19)$$

$$\mathbf{y}(k\Delta t) = \mathbf{C_d}\mathbf{x}(k\Delta t) + \mathbf{D_d}u(k\Delta t) \qquad (20)$$

where $\quad \mathbf{A_d} = e^{\mathbf{A_c}\Delta t}, \quad \mathbf{B_d} = (\int_0^{\Delta t} e^{\mathbf{A_c}\tau}d\tau)\mathbf{B_c} = [e^{\mathbf{A_c}\Delta t} - \mathbf{1}]\mathbf{A_c}^{-1}\mathbf{B_c}, \mathbf{C_c} = \mathbf{C_d}$, and $\mathbf{D_c} = \mathbf{D_d}$.

For TSSIT, the equations are

$$\mathbf{x}(k+1)\Delta t = \mathbf{A_d}\mathbf{x}(k\Delta t) + \mathbf{B_d}\mathbf{u}(k+1)\Delta t \qquad (21)$$

$$\mathbf{y}(k\Delta t) = \mathbf{C_d}\mathbf{x}(k\Delta t) + \mathbf{D_d}\mathbf{u}(k\Delta t). \qquad (22)$$

For a ramp-invariant input

$$\mathbf{u}(t) = \mathbf{u}(k\Delta t) + \frac{\mathbf{u}\left[(k+1)\Delta t\right] - \mathbf{u}(k\Delta t)}{\Delta t} \times (t - k\Delta t) \qquad (23)$$

where $k\Delta t \leq t \leq (k+1)\Delta t$, $k = 0, 1, 2, 3, \ldots$, and the output $\mathbf{y}(t)$ at $t = k\Delta t$ can be described as [19]

$$\mathbf{x}(k+1)\Delta t = \mathbf{F_d}\mathbf{x}(k\Delta t) + \mathbf{G_d}\mathbf{u}(k\Delta t) + \mathbf{H_d}\mathbf{u}(k+1)\Delta t$$

$$\mathbf{y}(k\Delta t) = \mathbf{C_d}\mathbf{x}(k\Delta t) + \mathbf{D_d}\mathbf{u}(k\Delta t) \qquad (24)$$

where

$$\mathbf{F_d} = e^{\mathbf{A_c}\Delta t} = \sum_{n=0}^{\infty} \frac{1}{n!}(\mathbf{A_c}\Delta t)^n \qquad (25)$$

$$\mathbf{G_d} = \left[e^{\mathbf{A_c}\Delta t}(-\mathbf{1} + \mathbf{A_c}\Delta t) + \mathbf{1}\right]\left(\mathbf{A_c}^2\Delta t^2\right)^{-1}\mathbf{B_c}\Delta t$$

$$= \sum_{n=0}^{\infty} \frac{1}{n(n+2)!}(\mathbf{A_c}\Delta t)^n \times \mathbf{B_c}\Delta t \qquad (26)$$

$$\mathbf{H_d} = \left[e^{\mathbf{A_c}\Delta t} - \mathbf{1} - \mathbf{A_c}\Delta t\right]\left(\mathbf{A_c}^2\Delta t^2\right)^{-1}\mathbf{B_c}\Delta t$$

$$= \sum_{n=0}^{\infty} \frac{1}{(n+2)!}(\mathbf{A_c}\Delta t)^n \times \mathbf{B_c}\Delta t \qquad (27)$$

$$\mathbf{C_d} = \mathbf{C_c} \qquad (28)$$

$$\mathbf{D_d} = \mathbf{D_c}. \qquad (29)$$

REFERENCES

[1] H. Li, M. Steurer, S. Woodruff, L. Shi, and D. Zhang, "Development of a unified design, test, and research platform for wind energy systems based on hardware-in-the-loop real-time simulation," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1144–1151, Jun. 2006.

[2] A. Bouscayrol, X. Guillaud, R. Teodorescu, P. Delarue, and W. Lhomme, "Energetic macroscopic representation and inversion-based control illustrated on a wind-energy-conversion system using hardware-in-the-loop simulation," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4826–4835, Dec. 2009.

[3] B. Lu, X. Wu, H. Figueroa, and A. Monti, "A low-cost real-time hardware-in-the-loop testing approach of power electronics control," *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, pp. 919–931, Apr. 2007.

[4] G. G. Parma and V. Dinavahi, "Real-time digital hardware simulation of power electronics and drives," *IEEE Trans. Power Del.*, vol. 22, no. 2, pp. 1235–1246, Apr. 2007.

[5] T. Zhou, B. Francois, M. Hadi Lebbal, and S. Lecoeuche, "Real-time emulation of a hydrogen-production process for assessment of an active wind-energy conversion system," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 737–746, Mar. 2009.

[6] D. Hercog, B. Gergic, S. Uran, and K. Jezernik, "A DSP-based remote control laboratory," *IEEE Trans. Ind. Electron.*, vol. 54, no. 6, pp. 3057–3068, Dec. 2007.

[7] A. Bouscayrol, "Different types of hardware-in-the-loop simulation for electric drives," in *Proc. IEEE ISIE*, Cambridge, U.K., Jun. 2008, pp. 2146–2151.

[8] T. L. Maguire and A. M. Gole, "Digital simulation of flexible topology power electronic apparatus in power systems," *IEEE Trans. Power Del.*, vol. 6, no. 4, pp. 1831–1840, Oct. 1991.

[9] V. R. Dinavahi, M. R. Iravani, and R. Bonert, "Real-time digital simulation of power electronic apparatus interfaced with digital controllers," *IEEE Trans. Power Del.*, vol. 16, no. 4, pp. 775–781, Oct. 2001.

[10] B. D. Kelper, L. A. Dessaint, K. A. Haddad, and H. Nakra, "A comprehensive approach to fixed-step simulation of switched circuits," *IEEE Trans. Power Electron.*, vol. 17, no. 2, pp. 216–224, Mar. 2002.

[11] V. Dinavahi, R. Iravani, and R. Bonert, "Design of a real-time digital simulator for a D-STATCOM system," *IEEE Trans. Ind. Electron.*, vol. 51, no. 5, pp. 1001–1008, Oct. 2004.

[12] K. Strunz, L. R. Linares, J. R. Marti, O. Huet, and X. Lombard, "Efficient and accurate representation of asynchronous network structure changing phenomena in digital real time simulators," *IEEE Trans. Power Syst.*, vol. 15, no. 2, pp. 586–592, May 2000.

[13] M. O. Faruque, V. Dinavahi, and W. Xu, "Algorithms for the accounting of multiple switching events in digital simulation of power electronic systems," *IEEE Trans. Power Del.*, vol. 20, no. 2, pp. 1157–1167, Apr. 2005.

[14] H. W. Dommel, "Digital computer solution of electromagnetic transients in single and multi-phase networks," *IEEE Trans. Power App. Syst.*, vol. PAS-88, no. 4, pp. 388–395, Apr. 1969.

[15] J. R. Marti and J. Lin, "Suppression of numerical oscillations in the EMTP," *IEEE Trans. Power Syst.*, vol. 4, no. 2, pp. 739–747, May 1989.

[16] N. R. Watson and J. Arrillaga, *Power Systems Electromagnetic Transients Simulation*. Stevenage, U.K.: IET, 2003.

[17] L. Naredo, A. Ramirez, A. Ametani, A. Gutierrez, A. Mansoldo, A. Gole, A. Lima, A. Morched, B. Gustavsen, D. Wilcox, F. Uribe, F. Moreira, F. de Leon, J. Martinez, L. Guardado, M. Davila, M. Ritual, N. Nagaoka, N. Watson, P. Gomez, P. Moreno, R. Iravani, S. Carneiro, T. Noda, V. Dinavahi, V. Ortiz, and W. Neves, "z-transform-based methods for electromagnetic transient simulation," *IEEE Trans. Power Del.*, vol. 22, no. 3, pp. 1799–1805, Jul. 2007.

[18] C.-T. Chen, *Analog and Digital Control System Design: Transfer-Function, State-Space, and Algebraic Methods*. Philadelphia, PA: Saunders, 1993.

[19] L. O. Chua and P. M. Lin, *Computer Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[20] L.-F. Pak, M. O. Faruque, X. Nie, and V. Dinavahi, "A versatile cluster-based real-time digital simulator for power engineering research," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 455–465, May 2006.

[21] *RT-LAB Manual*, OPAL-RT Technol. Inc., Montreal, QC, Canada, 2008.

[22] C. Schauder and H. Mehta, "Vector analysis and control of advanced static VAR compensators," *Proc. Inst. Elect. Eng.*, vol. 140, pt. C, no. 4, pp. 299–306, Jul. 1993.

M. Omar Faruque (SM'03–M'08) received the B.Sc.Engg. degree from Chittagong University of Engineering and Technology, Chittagong, Bangladesh, in 1992, the M.Engg.Sc. degree from the University of Malaya, Kuala Lumpur, Malaysia, in 1999, and the Ph.D. degree from the University of Alberta, Edmonton, AB, Canada, in 2008.

He is currently an Assistant Scholar/Scientist in The Center for Advanced Power Systems, The Florida State University, Tallahassee. His research interests are in the areas of hardware-in-the-loop simulation, modeling, and simulation of flexible ac transmission systems, HVDC, renewable resources, and grid connection of distributed resources and their impacts.

Venkata Dinavahi (S'94–M'00–SM'08) received the B.Eng. degree in electrical engineering from Visvesvaraya Regional College of Engineering, Nagpur, India, in 1993, the M.Tech. degree from the Indian Institute of Technology, 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 an Associate Professor at the University of Alberta, Edmonton, AB, Canada. His research interests include real-time simulation of power systems and power electronic systems, large-scale system simulation, and parallel and distributed computing.