

ORIGINAL RESEARCH

Open Access



FPGA-based real-time simulation for EV station with multiple high-frequency chargers based on C-EMTP algorithm

Zirun Li, Jin Xu^{*} , Keyou Wang, Pan Wu and Guojie Li

Abstract

The electric vehicle (EV) charging station is a critical part of the infrastructure for the wide adoption of EVs. Real-time simulation of an EV station plays an essential role in testing its operation under different operating modes. However, the large numbers of high-frequency power electronic switches contained in EV chargers pose great challenges for real-time simulation. This paper proposes a compact electromagnetic transient program (C-EMTP) algorithm for FPGA-based real-time simulation of an EV station with multiple high-frequency chargers. The C-EMTP algorithm transforms the traditional EMTP algorithm into two parallel sub-tasks only consisting of simple matrix operations, to fully utilize the high parallelism of FPGA. The simulation time step can be greatly reduced compared with that of the traditional EMTP algorithm, and so the simulation accuracy for high-frequency power electronics is improved. The EV chargers can be decoupled with each other and simulated in parallel. A CPU-FPGA-based real-time simulation platform is developed and the proposed simulation of the EV station is implemented. The control strategy is simulated in a CPU with 100 μ s time-step, while the EV station circuit topology is simulated in a single FPGA with a 250 ns time-step. In the case studies, the EV station consists of a two-level rectifier and five dual-active bridge (DAB) EV chargers. It is tested under different scenarios, and the real-time simulation results are validated using PSCAD/EMTDC.

Keywords: Dual-active bridge (DAB), Electromagnetic transient program (EMTP), Field-programmable gate arrays (FPGA), EV charger, Real-time simulation

1 Introduction

The electric vehicle (EV) is an effective way to tackle environmental challenges such as carbon emission [1, 2]. Due to limited battery capacity, increasing utilization of EVs requires widely installed charging stations [3]. Real-time simulation plays an essential role in the study of the electromagnetic transient characteristics of EV charging stations. These can accelerate the design of control and protection systems. In addition, real-time simulation with hardware-in-the-loop (HIL) simulation can realize joint simulations of an EV charging station model and an actual control prototype. This can reduce test costs

and shorten the development cycle compared with off-line simulation [4, 5]. However, simulating multiple high-frequency EV chargers real-time in small steps is a significant challenge due to the need for detailed device-level modeling of power electronic converters (PECs) and accurate interactions [6, 7]. Therefore, it is valuable to develop an efficient real-time simulation algorithm and implementation method for the system.

The typical time step of traditional real-time simulators is in the range of 20 μ s ~ 100 μ s, and interpolation algorithms are required to accurately reflect the switching events [8, 9]. However, the interpolation algorithms incur a great computing burden in real-time simulation. Therefore, in recent years, the field programmable gate array (FPGA) is used for small time step real-time

* Correspondence: xujin20506@sjtu.edu.cn

Key Laboratory of Control of Power Transmission and Conversion (SJTU),
Ministry of Education, Shanghai, China

simulation of detailed models because of its numerous logic resources [10–13], which can achieve high parallel computing. Thus, fast-switching circuit topology can be simulated in the FPGA while the control strategy is still performed in the CPU [14]. In this work, a lightweight CPU-FPGA architecture real-time simulation platform is established for simulating an EV station with multiple high-frequency EV chargers in a single FPGA.

There are currently two main categories of EMTP algorithm: state-space method [15, 16] and nodal analysis method [17], while both methods have been applied in real-time simulation. To achieve high accuracy and efficiency in device-level real-time simulation, a compact EMTP (C-EMTP) algorithm is proposed for the EV station simulated on FPGA. The C-EMTP algorithm compresses the serial computing process of a traditional EMTP into two parallel sub-tasks consisting of simple matrix calculation, to make full use of the high parallelism of FPGA. In addition, the C-EMTP algorithm avoids the calculation of intermediate variables such as injection current, and thus the computational complexity in each step is greatly reduced. The simulation time step is thus reduced and hardware resources are saved by optimizing the simulation process. In addition to optimizing the simulation algorithm, there have been some studies on the decoupling of the simulation circuit [18, 19]. A system decoupling method is proposed for the EV station in this paper. This decouples the EV station into multiple EV chargers to be simulated in parallel. Different from the traditional EMTP algorithm, using this method in C-EMTP can divide large matrix operations into small matrix operations, so as to further reduce the computing burden of FPGA. In addition to the C-EMTP algorithm, the associated discrete circuit (ADC) switch model [20, 21] is adopted to accurately reflect the switching moments to further improve the performance of the real-time simulation.

Using high-level synthesis tools for FPGA implementation can improve the efficiency of the simulation [22]. In addition to the lightweight hardware platform and the C-EMTP algorithm mentioned above, implementation methodologies are proposed to optimize the EV station simulation. Given the fast-switching characteristic of the EV chargers and computing complexity of the control strategy, the circuit is simulated at 250 ns time-steps while the control signals are updated every 100 μ s. The simulation loop operation of each subsystem can be pipelined to reduce resource consumption and latency. The idle time is eliminated to further reduce the time step of the simulation, while the use of fixed point matrix operation method saves hardware resources.

This paper is organized as follows. The structure of the real-time simulation system for the EV station is discussed in Section II. The C-EMTP algorithm and system

decoupling method are proposed in Section III, while Section IV gives the details of the implementation. The simulation study and resource consumption are discussed in Section V. These are verified by PSCAD/EMTDC simulations. Finally, the conclusion is drawn in Section VI.

2 The structure of the simulation system

The EV station, which is integrated with five EV chargers and one central rectifier, is chosen as the study case to perform on the real-time simulation platform. Each EV charger uses the dual active bridge (DAB) topology, while the central rectifier is a two-level converter. The main parameters are shown in Table 1. In this paper, the central rectifier maintains the DC bus voltage and each EV charger controls its battery charging power, as shown in Fig. 1.

The main architecture of the lightweight CPU-FPGA-based platform is shown in Fig. 2. It integrates the CPU and FPGA resources. The PXIe-8135 is the master board, which contains a quad-core Intel i7-3610QE processor, dual channel DDR3, 1600 MHz memory controller, all the standard I/O, and an integrated hard drive, and is connected with the Kintex-7 XC7K410T FPGA board and the host-PC. The control strategies are performed in the CPU, which receives the instantaneous circuit parameters from the FPGA and sends the modulated wave signals to the FPGA through the PXIe bus. The cores of the CPU run at 2.3GHz. The FPGA board is a slave board, which performs the real-time simulation of the EV station circuit and the generation of PWM switch signals under the instruction of the master board. The clock frequency of the FPGA board is set to 160 MHz. There are 254,200 look-up tables (LUTs) and 508,400 flip-flops (FFs) which constitute the configurable logic blocks (CLBs) to realize both the combinational logic and sequential logic. The FPGA contains 1540 digital signal processing (DSP) slices and 28,620 block

Table 1 Real-time simulation parameters

Parameter	Unit	Value
AC power grid V_g	V	220
Filter inductor of rectifier L_g and L_f	mH	1
Filter capacitor of rectifier C_f	μ F	100
Capacitor on DC bus C_{dc}	μ F	800
Leakage inductance of transformer L	μ H	160
Transformer ratio N_f	/	2:1
Output filter capacitor C_o	μ F	800
DC bus rated voltage V_{dc}	V	800
Output Voltage V_o	V	300–500
Output power P_o	kW	0–8
Rectifier switching frequency	kHz	10
DAB switching frequency	kHz	50

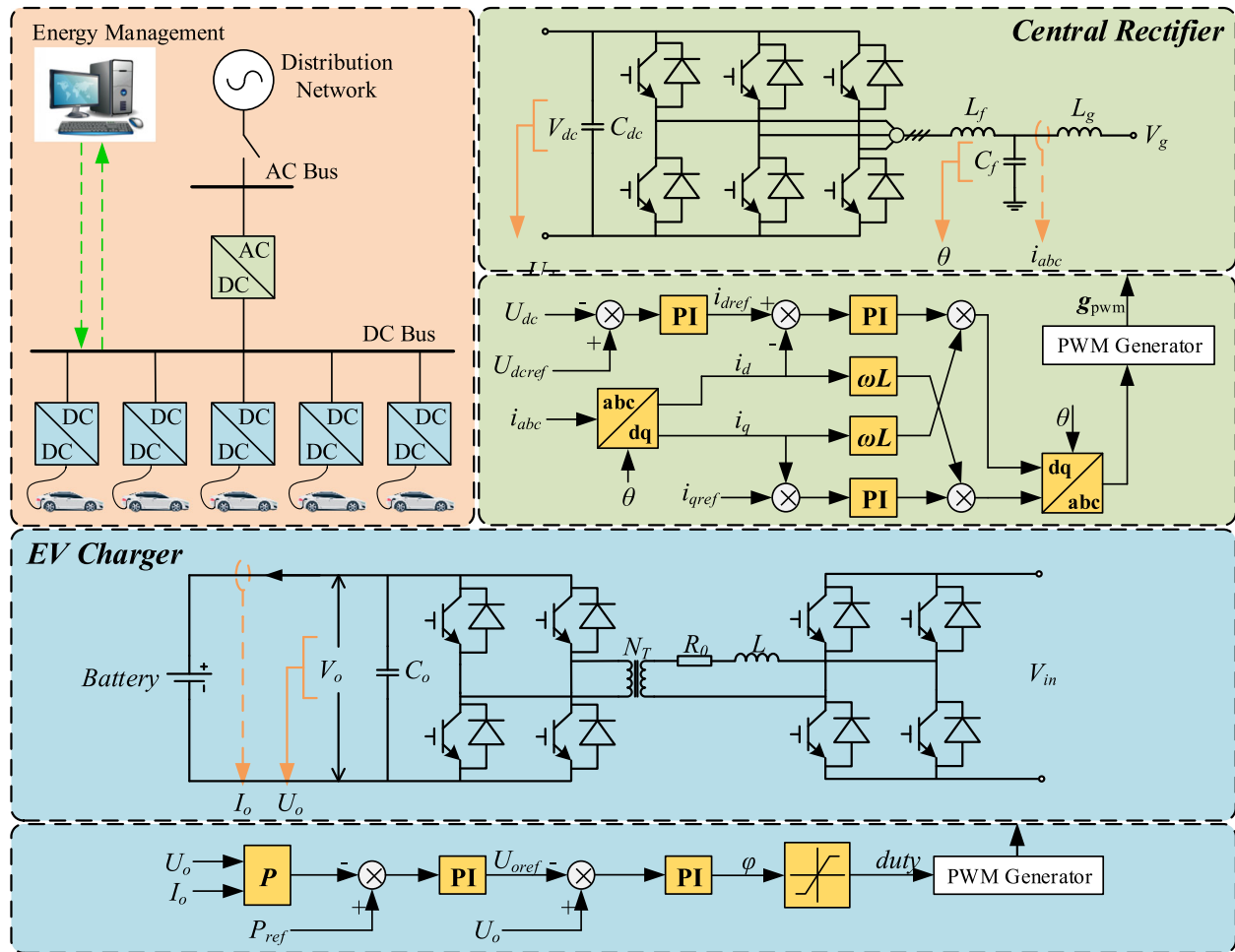


Fig. 1 EV charging system

RAMs, which provide 25×18 multipliers and memory resources mainly used in the matrix operation for the history current update and circuit parameter calculation. In addition, the FPGA I/O interfaces can be connected to oscilloscopes to observe the simulation results.

Communication between the CPU controller and the FPGA plays a vital role in real-time simulation. PXI

Express increases the available bandwidth to 8GB/s, enabling low-latency data exchange and improving the real-time simulation system performance, particularly when a sub-microsecond level time step is required. The host-PC provides a human-machine interface (HMI), receiving sampling data from the CPU and displaying the control and simulation waveforms.

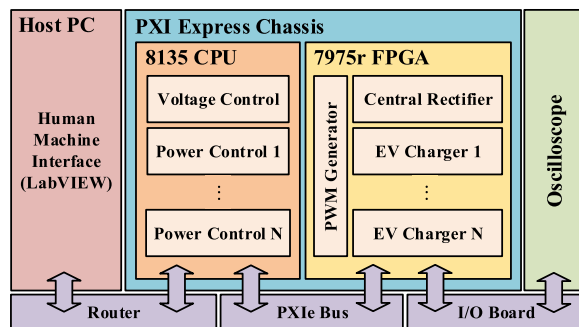


Fig. 2 The architecture of the real-time simulation platform for EV station

3 The simulation algorithm of EV Station

3.1 The C-EMTP algorithm

The traditional EMTP proposed by Professor Dommel is mainly based on a nodal analysis method, which involves complex serial calculations. The traditional EMTP algorithm also contains the calculation of intermediate variables such as injection current. However, only the node voltage and branch current in the simulation are required. In order to improve simulation efficiency, the C-EMTP algorithm is proposed so that the circuit parameters can be obtained directly by redesigning the simulation process. At the same time, the serial calculations

are transferred into two parallel sub-tasks, which only contain simple matrix calculations.

Similar to the traditional EMTP, all the branches in the network are transformed into Norton equivalent circuits. The following steps show the process of the simulation adopting the C-EMTP algorithm.

Step 1: The incidence matrix \mathbf{M}_{NAM} is formed in such a way that its matrix elements correspond to the nodes and branches in the circuit, as:

$$\mathbf{M}_{\text{NAM}} = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1N_b} \\ m_{21} & m_{22} & \cdots & m_{2N_b} \\ \vdots & \vdots & \ddots & \vdots \\ m_{N_n1} & m_{N_n2} & \cdots & m_{N_nN_b} \end{bmatrix}_{N_n \times N_b} \quad (1)$$

where N_n is the number of nodes and N_b is the number of branches in the circuit. When node i is connected to branch j and the current of branch j flows away from node i , the entry of \mathbf{M}_{NAM} (m_{ij}) is 1. On the other hand, when the current of branch j flows away from node i , the entry of \mathbf{M}_{NAM} (m_{ij}) is -1. When node i and branch j have no connection in the network, \mathbf{M}_{NAM} (m_{ij}) is 0.

Step 2: Voltage source or current source branches are represented by equivalent admittance and parallel current sources, while other branches are represented by equivalent admittance and a parallel history current source. The history current of each branch can be expressed by the branch voltage and current as:

$$I_h^{n+1} = \alpha Y_b V_b^n + \beta I_b^n \quad (2)$$

where Y_b is the equivalent admittance of the branch, α is the voltage coefficient and β is the current coefficient.

If the backward Euler method is applied for numerical integration, the equivalent admittance of all branches can be obtained. For the inductance branch L , there are:

$$\alpha = 0, \quad \beta = 1, Y_b = dt/L \quad (3)$$

For the capacitance branch C , there are:

$$\alpha = -1, \quad \beta = 0, Y_b = C/dt \quad (4)$$

For the switch branch, α and β vary with the switch state. The switch is equivalent to a small inductance L_s when it is on, and to a small capacitance C_s when it is off. Thus, according to (3) and (4), there are:

$$\begin{cases} \alpha = 0, \quad \beta = 1, Y_b = dt/L_s, & \text{ON state} \\ \alpha = -1, \quad \beta = 0, Y_b = C_s/dt, & \text{OFF state} \end{cases} \quad (5)$$

In order to improve the efficiency of simulation, an appropriate inductance L_s and capacitance C_s are chosen to keep Y_b constant, as:

$$Y_b = dt/L_s = C_s/dt \quad (6)$$

In this step, only α and β need to be updated during the simulation (Fig. 3).

Rewriting (2) to matrix form yields:

$$\mathbf{I}_h^{n+1} = \alpha \mathbf{Y}_b \mathbf{V}_b^n + \beta \mathbf{I}_b^n \quad (7)$$

where \mathbf{I}_h , \mathbf{V}_b , and \mathbf{I}_b are the $N_b \times 1$ vectors consisting of I_h , V_b , and I_b of each branch, respectively. \mathbf{Y}_b is the $N_b \times N_b$ diagonal matrix composed of Y_b , whereas α and β are the $N_b \times N_b$ diagonal matrices consisting of α and β , respectively.

Step 3: According to the history current \mathbf{I}_h and equivalent current \mathbf{I}_s of voltage/current source, there is

$$\mathbf{I}_{inj}^{n+1} = -\mathbf{M}_{\text{NAM}} (\mathbf{I}_h^{n+1} + \mathbf{I}_s^{n+1}) \quad (8)$$

where \mathbf{I}_{inj} is the $N_b \times 1$ vector composed of the injection current. Then, the nodal voltage vector \mathbf{V}_n , the branch voltage vector \mathbf{V}_b , and the branch current vector \mathbf{I}_b can be expressed as:

$$\mathbf{V}_n^{n+1} = \mathbf{Y}_n^{-1} \mathbf{I}_{inj}^{n+1} \quad (9)$$

$$\mathbf{V}_b^{n+1} = \mathbf{M}_{\text{NAM}}^T \mathbf{V}_n^{n+1} \quad (10)$$

where \mathbf{Y}_n is the $N_n \times N_n$ nodal admittance matrix. By analyzing the relationship between branch voltage and current, the following equation can be obtained:

$$\mathbf{I}_b^{n+1} = \mathbf{Y}_b \mathbf{V}_b^{n+1} + \mathbf{I}_h^{n+1} + \mathbf{I}_s^{n+1} \quad (11)$$

Equations (7)–(11) reveal that the simulation requires many serial steps. When implemented on hardware, the intermediate variables of the traditional EMTP algorithm consume large resources and reduce efficiency. Therefore, a compact simulation loop is derived as:

$$\begin{cases} \mathbf{V}_n^n = -\mathbf{Y}_n^{-1} \mathbf{M}_{\text{NAM}} (\mathbf{I}_h^n + \mathbf{I}_s^n) = \mathbf{K} (\mathbf{I}_h^n + \mathbf{I}_s^n) \\ \mathbf{I}_b^n = (-\mathbf{Y}_b \mathbf{M}_{\text{NAM}}^T \mathbf{Y}_n^{-1} \mathbf{M}_{\text{NAM}} + \mathbf{I}) (\mathbf{I}_h^n + \mathbf{I}_s^n) = \mathbf{J} (\mathbf{I}_h^n + \mathbf{I}_s^n) \end{cases} \quad (12)$$

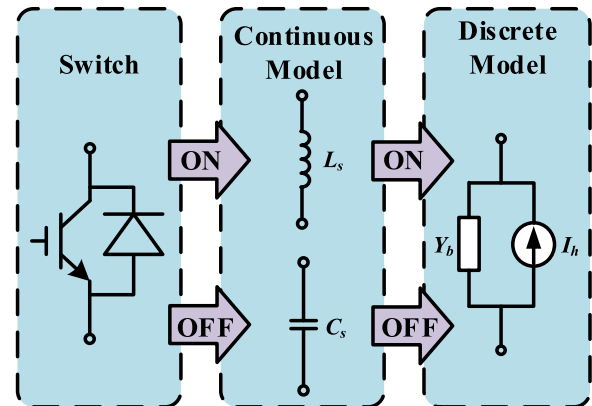


Fig. 3 Norton equivalent circuit of the switch

where K and J are coefficient matrices, which can be pre-calculated and pre-stored. Thus, the C-EMTP algorithm saves the resources occupied by intermediate variables during the simulation loop. This is beneficial for the simulation speed while at the same time may also reduce the error diffusion.

In the simulation loop, the node voltage and branch current are decided purely by the last time step history current and source equivalent current. Recording the index of node number and branch number such that the first N_n nodes are node voltage and the latter N_b branches are branch current, the following matrix equation can be obtained:

$$\begin{bmatrix} \mathbf{V}_n^n \\ \mathbf{I}_b^n \end{bmatrix} = \begin{bmatrix} \mathbf{K} \\ \mathbf{J} \end{bmatrix} (\mathbf{I}_h^n + \mathbf{I}_s^n) \quad (13)$$

$$\mathbf{I}_h^{n+1} = (\alpha + \beta) \cdot \mathbf{J} \cdot (\mathbf{I}_h^n + \mathbf{I}_s^n) - \alpha (\mathbf{I}_h^n + \mathbf{I}_s^n) \quad (14)$$

The simulation loop is more compact and the number of penalties required in a single step is reduced compared to the traditional EMTP method.

3.2 System decoupling method

The EV station with multiple high-frequency EV chargers consists of detailed converter models, especially the switching frequency of EV chargers is 50 kHz. Thus, in order to guarantee that the simulation loop can be completed in a time step, a system decoupling method is adopted to further improve the real-time simulation performance. This method decouples the system into multiple EV chargers, which can interact through the interface. This method realizes the parallel simulation of each EV charger and makes full use of the parallelism of FPGA. In comparison, the non-decoupled system involves large-scale non-diagonal matrix operations, which limit the minimum simulation time step and consume more hardware resources to execute. Dividing a system into multiple subsystems can effectively reduce the complexity of matrix operations.

The capacitor C is selected as the interface between multiple subsystems as shown in Fig. 4. It is equivalent to two same voltage sources in branch p and branch q . Then, the interface voltage can be written as:

$$\begin{cases} V_n^{n+1} = dV_n^n + V_n^n \\ dV_n^n = [i_s^n(p) + i_s^n(q)] \cdot dt / C \end{cases} \quad (15)$$

The error analysis of the decoupling method is given in the [Appendix](#). The third-order local truncation error caused by the decoupling method is relatively small compared to the non-decoupling method and the influence on the results is negligible.

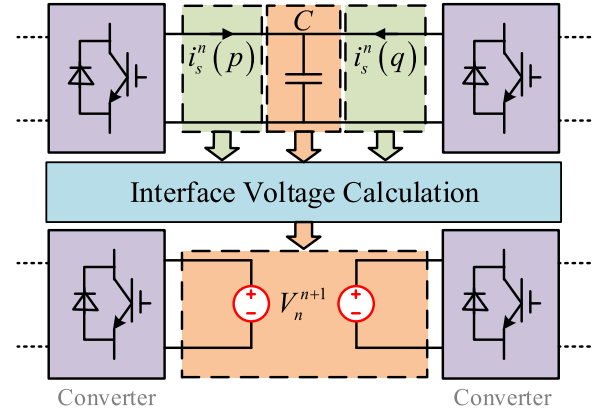


Fig. 4 System decoupling method

Based on the C-EMTP algorithm, the real-time simulation procedure is shown in Fig. 5. The green part shows the initialization of the electrical circuit, in which each subsystem performs its own initialization and the coefficient matrices and variables are loaded to the corresponding simulation steps. The subsystems are updated according to (15) in the blue part, while the blue part and yellow part indicate a closed loop between the circuit and control. The node voltage V_n and brand current I_b are sent to the CPU and the control signals are transferred to α and β to control the circuit. The purple part shows the interface calculation between the subsystems.

The whole EV station circuit is modeled in one single FPGA board for maximum resource utilization. This method avoids the amount of data to be exchanged between boards and reduces simulation latency. Normally, the subsystems are decoupled at the capacitor, and thus each EV charger composes a subsystem and the central rectifier is also defined as a subsystem. With this decoupling method, the scalability of the electric vehicle charging station simulation is improved, and the number of simulated EV chargers can be adjusted at any time according to actual needs.

4 Platform implementation

4.1 Simulation loop

To improve the efficiency of simulating a larger EV station in one FPGA board, a CPU-FPGA-based real-time simulation platform is constructed in the National Instruments (NI) PXI chassis, in which the control strategy and circuit topology are simulated at different time steps. Figure 6 shows the system simulation loop between the electrical and control systems. This indicates the cooperation between the FPGA and CPU.

For the EV station, all the calculations of the circuit and control subsystems start simultaneously. The control signal calculation on the CPU usually takes more time than the circuit update on the FPGA depending on

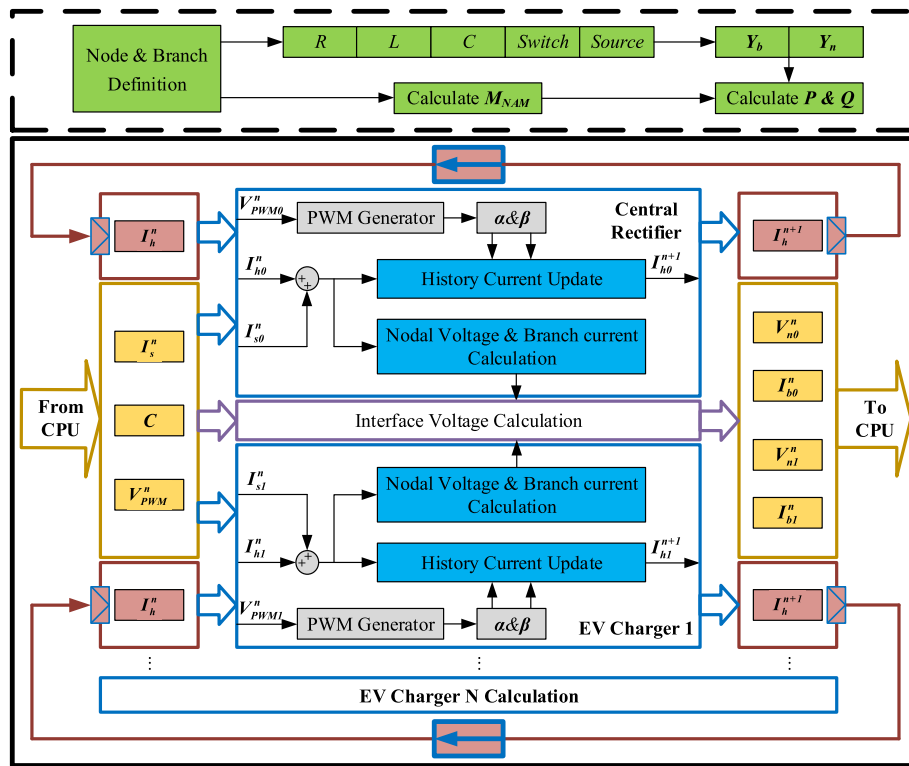


Fig. 5 The simulation loop of the EV charging station

the actual execution times of the CPU and FPGA. As a result, the FPGA must wait for the CPU until the next time step is reached. In order to improve real-time simulation performance, the control signal calculation in the CPU has a 100 μ s time-step and the circuit simulation is calculated in the FPGA with a 250 ns time-step. In this way, the FPGA and CPU operations are relatively independent while the data exchange is performed every 100 μ s. When the control signals are not updated, the FPGA runs the next simulation cycle with the control signals of the previous calculation.

When a simulation loop starts, the FPGA reads the control signal and the CPU reads the electrical variables of the last step. Then, all the modules of the electrical calculation begin simultaneously. At the same time, the CPU begins to update the control signal. During each time step in the FPGA, the node voltages, branch currents, history current sources and switch status are updated. By adopting the pipeline method, not only can each subsystem be calculated in parallel, but the calculation within the subsystem can also be performed in parallel, which further reduces the simulation time step in the FPGA.

4.2 FPGA implementation

The whole simulation implementation in the FPGA is shown in Fig. 7. As shown, data exchanges between the

CPU and FPGA are through FIFO. The FPGA receives the modulation wave signal and voltage or current source signal of the circuit from the CPU, while transmitting calculation results of the node voltage and branch current to the CPU. Since the control signal calculation on the CPU and circuit simulation on the FPGA are simulated with different time steps, the FIFO performs data exchange to realize asynchronous communication to ensure accurate interaction between the FPGA and CPU (Fig. 7).

In the FPGA, the C-EMTP algorithm simulates the electrical system, which mainly contains the update of the switch states, the history current and the calculation of the node voltage and the branch current. The system can be solved by floating-point operation in offline simulation to obtain high accuracy results. However, it is more suitable to adopt fixed-point operation to satisfy the clock requirement and limit FPGA multiplier resources. The fixed-point format of the variables is set to $\langle \pm, 25, 12 \rangle$, which is confirmed by the DSP multiplier. A DSP slice is used for the 25bits \times 18bits multiplication, and the $\langle \pm, 25, 12 \rangle$ fixed-point multiplication uses two DSP slices that are fewer than in the single-precision floating-point format.

While the switch states are being updated to regenerate α and β , the multiplier reads the matrix K and J pre-stored in block RAM for the next calculation. In

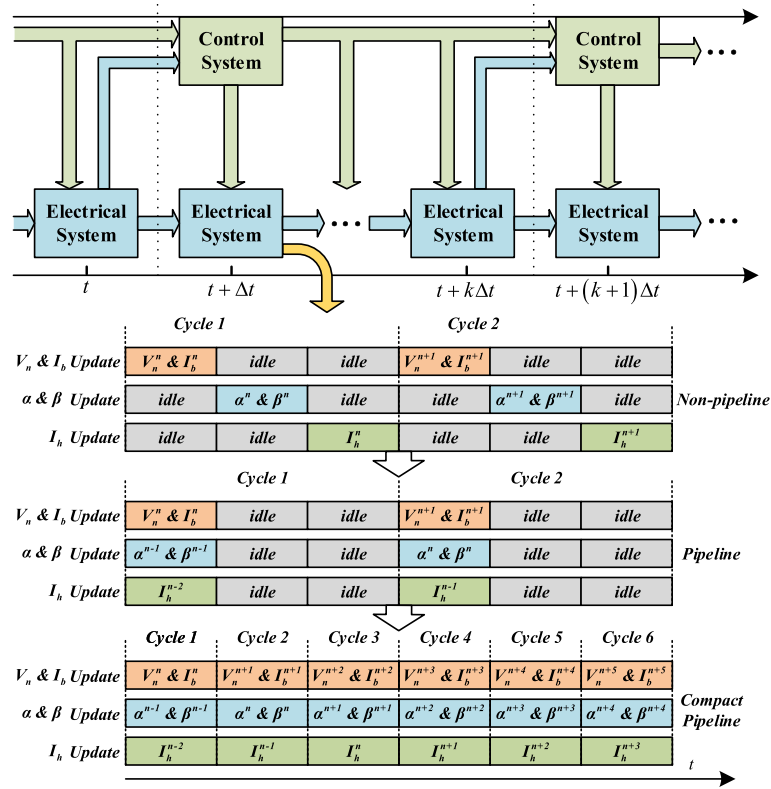


Fig. 6 Cooperation between FPGA and CPU

calculating the node voltage and branch current, mainly $25\text{bits} \times 18\text{bits}$ matrix-vector multiplications using DSP multipliers are performed. Matrix multiplications occupy the most resources and time in FPGA-based real-time simulation, while the multiplier consumption of non-

diagonal matrix multiplications is linearly proportional to the matrix order. According to the proposed system decoupling method, large-scale matrices existing in the non-decoupled EV station simulation are divided into small-scale matrices to avoid complex serial

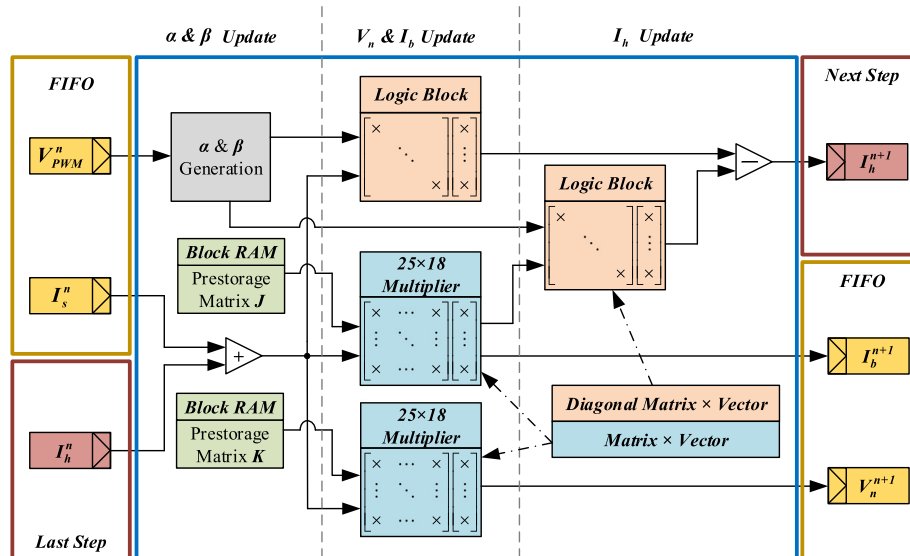


Fig. 7 The FPGA implementation of the C-EMTP algorithm

multiplications. For example, the EV station contains 48 nodes and 73 buses, which will generate a 73×73 matrix to be solved. Using the system decoupling method, the 73×73 matrix can be divided into one 13×13 matrix and five 12×12 matrices, which can be solved in parallel. The multiplications involving non-diagonal matrices are achieved by a multiply-accumulate module, which is executed in serial and spends the most clock cycles in each time step. The pipeline method is also adapted to execute the multiplication and accumulation. This saves nearly half of the execution time in each step.

To update the history current, mainly logic blocks are used for operations. Since α and β representing the switch states are both 1, 0 or -1 , logic resources are used for calculations to replace multipliers. This improves the history current update speed and saves multiplier resources.

5 Simulation results and discussion

The tested EV station consists of a two-level rectifier and five high-frequency DAB EV chargers. This section applies the C-EMTP algorithm to the CPU-FPGA-based real-time simulation platform to simulate the EV station. The results of the real-time simulation and corresponding PSCAD simulation are displayed in Fig. 8.

5.1 Resource consumption

According to the FPGA implementation details, the execution time, as well as the hardware resource utilization of each subsystem, are presented in Table 2. Look-up tables (LUTs), flip-flops (FF), digital signal processing (DSP) slices and block RAMs are recorded as the mainly consumed resources.

As discussed in Sections 3 and 4, the latencies of the additional steps in each subsystem are the same, and only the inconsistency of matrix dimensions leads to different latency and resource consumption. The number of rows of the matrix determines the DSP slice needed for calculation, and the number of matrix columns determines the time required for matrix multiplication. For example, the central rectifier subsystem that contains 8 nodes and 13 branches consumes the most simulation time and hardware resource because the matrices P and Q have a larger size (21×13) than other subsystems (20×12).

In the EV station real-time simulation, the execution time is constant which is consistent with the central rectifier subsystem. It demonstrates that the latency is determined by the central rectifier subsystem, which has the largest maximum latency of 38 clock cycles. When new EV chargers are integrated, the execution time will not increase as long as the scale of the new subsystem is not larger than the previous ones. If the traditional EMTP method is used, the real-time simulation can only

be realized with a simulation step of $1 \mu\text{s}$. However, when the electric vehicle charging station is working at a switching frequency of 50 kHz, the simulation step must be 250 ns or less to achieve the required simulation accuracy. Therefore, the C-EMTP algorithm must be used. In addition, to attain a smaller time step, the scales of the subsystems are similar in case of an increase of the system latency. The parallel structure ensures that more EV chargers can be integrated to take full advantage of the FPGA resources without time constraints.

For comparison, the central rectifier is simulated using the C-EMTP algorithm and the traditional algorithm, and the latencies of the simulations are shown in Table 3. It can be seen that the latency of the C-EMTP algorithm is much smaller than the traditional EMTP algorithm, proving that the C-EMTP algorithm greatly improves simulation efficiency. When trying to simulate the EV station with the traditional EMTP algorithm, the execution time exceeds $2 \mu\text{s}$, which is too large for the DAB switched at 50 kHz. However, when the EV station is simulated with the C-EMTP algorithm, the 250 ns time-step real-time simulation is realized. Since the system clock is 160 MHz, it means that the circuit simulation of the EV station in one time step is accomplished within 40 clock cycles.

5.2 Plug-and-play scenarios

The EV charger is designed to achieve plug-and-play capability and can adjust the output power according to different vehicles. This greatly improves the convenience of charging. Figure 8 shows the transient waveforms of the plug and play scenarios. EV chargers 4 and 5 start charging the EVs at the power of 4 kW and 5 kW at 2 s and the charging power of EV charger 1 changes from 8 kW to 3 kW at 3 s. At 4 s, EV chargers 3 and 5 are cut off from the charging station. When there is an abrupt power change, both the AC and DC buses experience disturbances. The charging current of the EV station and input DC bus voltage of the EV chargers observed from the oscilloscope are shown in Fig. 8 (a1) and (a2), respectively. Since the central rectifier is responsible for maintaining the DC bus voltage, the DC voltage returns to the rated voltage quickly after each disturbance. Figure 8 (a3) shows the power tracing performance of the EV chargers observed from the host PC, and all follow the control signals well. From Fig. 8 (a1)-(a3) it can be seen that the output power of the EV chargers track the power commands instantly after the plug and disconnection of the EVs. When the charging power changes, the input current changes accordingly. The off-line simulation results in PSCAD/EMTDC are conducted for validation as shown in Fig. 8 (b1)-(b3).

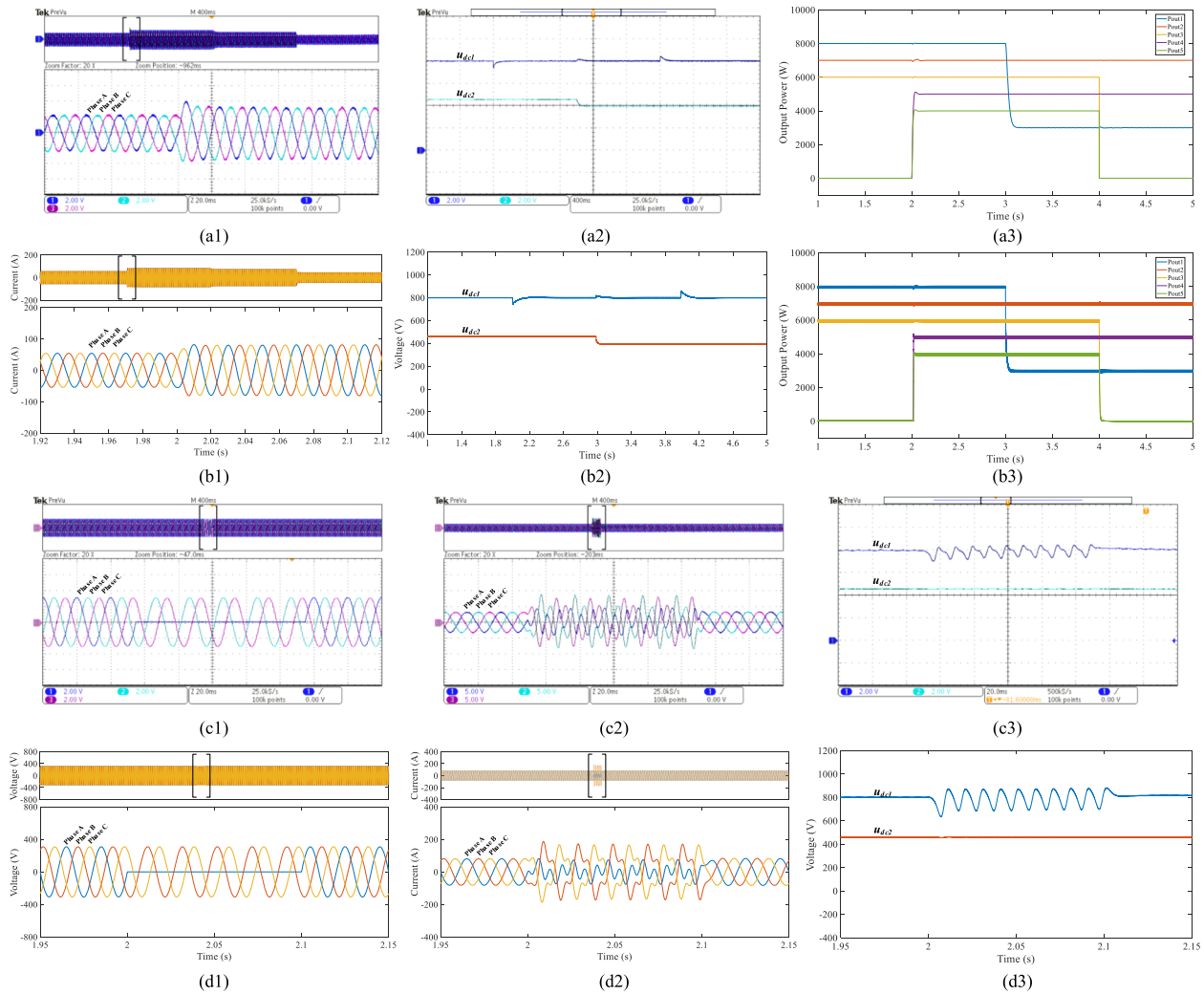


Fig. 8 Transient waveforms of the EV charging station under different conditions

5.3 Fault operation scenarios

A single-phase to ground fault is applied to the AC bus at 2.0 s of the simulation with the steady-state initial condition and is cleared after 0.1 s. The transient waveforms of the AC bus voltage and current and DC bus voltage are presented in Fig. 8(c1)–(c3). The AC side operates under an asymmetric operating condition and the DC bus voltage fluctuates significantly during the fault. As the charging voltage is isolated by the DAB from the fault location, the voltage is only slightly affected so the impact of faults on EVs is effectively reduced. When the fault is cleared, the DC bus voltage is restored to the rated voltage and the EV chargers go back to normal operation mode. It indicates that the EV

station has the ability to ride through the fault. The fault transient results in PSCAD/EMTDC are shown in Fig. 8 (d1)–(d3). Through the real-time simulation with the C-EMTP algorithm, the fault transient waveform can be obtained and the control strategies for the EV station can be verified.

6 Conclusion

In this paper, a C-EMTP algorithm has been proposed to achieve real-time simulation of an EV station in a lightweight CPU-FPGA-based platform. The simulation results of the C-EMTP algorithm for an EV station with multiple high-frequency EV chargers prove to have a

Table 2 Hardware resource consumption of EV station

Device	FF	LUT	RAM	DSP48
XC7K410T	315,555 (62.1%)	208,527 (82.0%)	91 (11.4%)	1078 (70.0%)

Table 3 Time latency of simulation algorithm

Parameter	Latency
C-EMTP Algorithm	38 ticks (237.5 ns)
Traditional EMTP Algorithm	118 ticks (737.5 ns)

good consistency with the simulation results in PSCAD with a switching frequency of 50 kHz. The time latency study shows that compared with the traditional EMTP algorithm, the C-EMTP algorithm for the EV station reduces the simulation execution time by more than 65%, which means that a small time step of 250 ns can be used. The system decoupling method ensures that the time step remains constant when the scale of the simulation increases. The proposed C-EMTP algorithm and platform implementation in high switching-frequency simulation has great potential for high-precision simulation of power systems with high converter penetration.

7 Appendix

The error of the decoupling method in Section 3.2 has negligible effect on the simulation results. According to the decoupling method in Fig. 4, the capacitor voltage is expressed as:

$$\begin{aligned} u_E(t + \Delta t) &= u_E(t) + [i_p(t) + i_q(t)] \cdot \frac{\Delta t}{C} \\ &= u_E(t) + I_{inj}(t) \cdot \frac{\Delta t}{C} \end{aligned}$$

It can be seen from the above equation that the capacitor is discretized by the Euler method. At the same time, the rest of the circuit still uses the backward Euler method.

For the Euler method, the local truncation error is:

$$\begin{aligned} T_E^{n+1} &= u_E(t + \Delta t) - u_E(t) - \Delta t \cdot f(t, u_E(t)) \\ &= \frac{\Delta t^2}{2} u_E''(t) + O(\Delta t^3) = O(\Delta t^2) \end{aligned}$$

If the decoupling method is not used, the whole circuit adopts the backward Euler method and the capacitor voltage is expressed as:

$$\begin{aligned} u_B(t + \Delta t) &= u_B(t) + [i_p(t + \Delta t) + i_q(t + \Delta t)] \cdot \frac{\Delta t}{C} \\ &= u_B(t) + I_{inj}(t + \Delta t) \cdot \frac{\Delta t}{C} \end{aligned}$$

For the backward Euler method, the local truncation error is:

$$\begin{aligned} T_B^{n+1} &= u_B(t + \Delta t) - u_B(t) - \Delta t \cdot f(t + \Delta t, u_B(t + \Delta t)) \\ &= -\frac{\Delta t^2}{2} u_B''(t) + O(\Delta t^3) = O(\Delta t^2) \end{aligned}$$

The local truncation errors relative to the analytical solution under both methods are:

$$T_e = O(\Delta t^2)$$

It can be seen from the above analysis that the use of the decoupling method has little effect on the system

simulation error, while both methods have first-order accuracy.

The local truncation error of the decoupling method relative to the non-decoupling method is:

$$T'_e = u_E(t + \Delta t) - u_B(t + \Delta t) = O(\Delta t^3)$$

In the same way, the error caused by the decoupling method is negligibly small compared to the non-decoupling method.

Abbreviations

EV: Electric Vehicle; C-EMTP: Compact Electromagnetic Transient Program; FPGA: Field-programmable Gate Arrays; PEC: Power Electrical Converters; LUT: Look-up Table; FF: Flip-flop; CLB: Configurable Logic Block; DSP: Digital Signal Processing; HMI: Human-machine Interface; NI: National Instruments; DAB: Dual-active Bridge

Acknowledgements

This work is supported by China Postdoctoral Science Foundation (BX20200221, 2020 M671122), National Key Research and Development Program of China (2019YFE0122600), National Natural Science Foundation of China (51877133).

About the authors

Z. R. Li(1996-), male, PHD student, Major in modeling and real-time simulation of power system integrated with renewable energy.

E-mail: lzr9602@sjtu.edu.cn.

J. Xu(1991-), male, PHD and Post-doctor, Major in power system stability analysis, power electronic modeling, and real-time simulation.

E-mail: xujin20506@sjtu.edu.cn.

K. Y. Wang(1979-), male, PHD and Professor, Major in power system dynamic and stability, renewable energy integration, and converter dominated power systems.

E-mail: wangkeyou@sjtu.edu.cn.

P. Wu(1995-), male, PHD student, Major in control, modeling and real-time simulation of renewable energy.

E-mail: panghuwu@sjtu.edu.cn.

G. J. Li(1965-), male, PHD and Professor, Major in power system analysis and control, wind and PV power control and integration, and microgrid.

E-mail: liguojie@sjtu.edu.cn.

Authors' contributions

Zirun Li performed the study of the algorithm, verified the simulation and draft the manuscript. Jin Xu, Keyou Wang and Guojie Li engaged in modifying the paper and submitted it to the PCMP. Pan Wu participated in the real-time simulation experiments. All authors read and approved the final manuscripts.

Funding

This work is supported by China Postdoctoral Science Foundation (BX20200221, 2020 M671122), National Key Research and Development Program of China (2019YFE012784), National Natural Science Foundation of China (51877133).

Availability of data and materials

Not Applicable.

Competing interests

No competing interests.

Received: 24 May 2020 Accepted: 23 October 2020

Published online: 27 November 2020

References

1. Battapothula, G., Yammani, C., & Maheswarapu, S. (2019). Multi-objective simultaneous optimal planning of electrical vehicle fast charging stations and DGs in distribution system. *Journal of Modern Power Systems and Clean Energy*, 7(4), 923–934.

2. Liu, H., Huang, K., Yang, Y., et al. (2018). Real-time vehicle-to-grid control for frequency regulation with high frequency regulating signal. *Protection and Control of Modern Power Systems*, 3, 13.
3. Bayati, M., Abedi, M., Gharehpetian, G., et al. (2019). Short-term interaction between electric vehicles and microgrid in decentralized vehicle-to-grid control methods. *Protection and Control of Modern Power Systems*, 4, 5.
4. Yu, Z., Chen, S., & Tong, L. (2016). An intelligent energy management system for large-scale charging of electric vehicles. *CSEE Journal of Power and Energy Systems*, 2(1), 47–53.
5. Wang, L., Gao, H., & Zou, G. (2017). Modeling methodology and fault simulation of distribution networks integrated with inverter-based DG. *Protection and Control of Modern Power Systems*, 2, 31.
6. Iyer, V. M., Guler, S., & Bhattacharya, S. (2019). Small-signal stability assessment and active stabilization of a bidirectional battery charger. *IEEE Transactions on Industry Applications*, 55(1), 563–574.
7. Xu, J., Wang, K., Li, G., Ji, W., Jiang, X., & Zhang, H. (2018). System-level dynamic phasor models of hybrid AC/DC microgrids suitable for real-time simulation and small signal analysis. *IET Generation Transmission and Distribution*, 12(15), 3607–3617.
8. Zhang, Y., Ding, H., & Kuffel, R. (2017). Key techniques in real time digital simulation for closed-loop testing of HVDC systems. *CSEE Journal of Power and Energy Systems*, 3(2), 125–130.
9. Dagbagi, M., Hemdani, A., Idkhajine, L., Naouar, M. W., Monmasson, E., & Slama-Belkhdja, I. (2016). ADC-based embedded real-time simulator of a power converter implemented in a low-cost FPGA: Application to a fault-tolerant control of a grid-connected voltage-source rectifier. *IEEE Transactions on Industrial Electronics*, 63(2), 1179–1190.
10. Shen, Z., Duan, T., & Dinavahi, V. (2018). Design and implementation of real-time Mpsoc-FPGA-based electromagnetic transient emulator of CIGRÉ DC grid for HIL application. *IEEE Power and Energy Technology Systems Journal*, 5(3), 104–116.
11. Matar, M., & Iravani, R. (April 2010). FPGA implementation of the power electronic converter model for real-time simulation of electromagnetic transients. *IEEE Transactions on Power Delivery*, 25(2), 852–860.
12. Myaing, A., & Dinavahi, V. (2011). FPGA-based real-time emulation of power electronic systems with detailed representation of device characteristics. *IEEE Transactions on Industrial Electronics*, 58(1), 358–368.
13. Wu, P., Wang, K., Xu, J., & Li, G. (2020). Real-time simulation algorithm design of a virtual synchronous grid-connected inverter system based on a CPU-FPGA heterogeneous platform. *Power System Protection and Control*, 48(14), 85–94.
14. Wang, K., Xu, J., Li, G., Tai, N., Tong, A., & Hou, J. (2019). A generalized associated discrete circuit model of power converters in real-time simulation. *IEEE Transactions on Power Electronics*, 34(3), 2220–2233.
15. Dufour, C., Mahseredjian, J., & Belanger, J. (2011). A combined state-space nodal method for the simulation of power system transients. *IEEE Transactions on Power Delivery*, 26(2), 928–935.
16. Dong, Y., Pan, L., Qiu, D., Tian, J., Wang, W., & Li, W. (2016). Hardware-in-the-loop simulation and test of a control and protection system for MMC-based UPFC. 2016 IEEE Power and Energy Society General Meeting (PESGM), Boston, MA, pp 1–5.
17. Chung-Wen, H., Ruehli, A., & Brennan, P. (1975). The modified nodal approach to network analysis. *IEEE Transactions on Circuits and Systems*, 22(6), 504–509.
18. Mu, Q., Liang, J., Zhou, X., Li, G., & Zhang, X. (2018). A node splitting interface algorithm for multi-rate parallel simulation of DC grids. *CSEE Journal of Power and Energy Systems*, 4(3), 388–397.
19. Duan, T., Shen, Z., & Dinavahi, V. (2019). Multi-rate mixed-solver for real-time nonlinear electromagnetic transient emulation of AC/DC networks on FPGA-MPSoC architecture. *IEEE Power and Energy Technology Systems Journal*, 6(4), 183–194.
20. Xu, J., Wang, K., Wu, P., & Li, G. (2020). FPGA-based sub-microsecond-level real-time simulation for microgrids with a network-decoupled algorithm. *IEEE Transactions on Power Delivery*, 35(2), 987–998.
21. Hui, S. Y. R., & Morrell, S. (1994). Generalised associated discrete circuit model for switching devices. *IEE Proceedings: Science, Measurement & Technology*, 141(1), 57–64.
22. Jiménez, Ó., Lucía, Ó., Urriza, I., Barragan, L. A., Navarro, D., & Dinavahi, V. (2015). Implementation of an FPGA-based online hardware-in-the-loop emulator using high-level synthesis tools for resonant power converters applied to induction heating appliances. *IEEE Transactions on Industrial Electronics*, 62(4), 2206–2214.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)