FPGA-Based Real-Time Simulation of Microgrids with a State-Space Splitting Approach

Xiaopeng Fu, Jiaming Wang, Peng Li*, Jiang Yu, Chengshan Wang
State Key Laboratory of Smart Power Distribution Equipment and System
Tianjin University
Tianjin, China
{fuxiaopeng, wjm, lip, yujiang2000, cswang}@tju.edu.cn

***Abstract*—With the rapid development of microgrids and the increasing integration of multiple power converters, higher requirements have been raised for the accuracy and efficiency of real-time electromagnetic transient (EMT) simulation. The high-frequency switching characteristics of power electronic devices lead to frequent topology changes, imposing severe challenges on real-time simulators in terms of small time-step computation, parallel solving, and memory utilization. To address these issues, this paper proposes a real-time EMT simulation method based on state-space splitting. The proposed method performs delay-free decoupling of the system matrix at the numerical computation level, dividing it into constant and time-varying components. By applying the exponential integrator, the method effectively reduces the computational burden and memory demand caused by high-frequency switching. Furthermore, leveraging the high parallelism of field-programmable gate arrays (FPGAs), a hierarchical computation framework is designed with hardware reuse and pipelining mechanisms optimized for dense/sparse matrix operations. This enables efficient algorithm-to-hardware co-design and establishes an FPGA based real-time EMT simulation framework for microgrids. Simulation results demonstrate that the proposed method significantly reduces computational load and improves hardware resource utilization while maintaining high accuracy, providing an effective approach for real-time simulation of complex microgrids.**

***Index Terms*—Electromagnetic transient, splitting method, exponential integrator, real-time simulation.**

I. INTRODUCTION

Real-time electromagnetic transient simulation is an essential technique for analyzing microgrids with multiple converters and enabling cost-effective hardware-in-the-loop (HIL) testing for control design and rapid prototyping [1]. However, the high switching frequency, fast transients, and nonlinearity of power electronic converters significantly increase computational demands, necessitating smaller time steps and driving ongoing research in efficient modeling, algorithmic acceleration, and hardware design [2].

In the field of EMT simulation, two fundamental solution frameworks are commonly used: the nodal analysis method and the state-space method. The nodal method is intuitive and widely adopted, but in microgrids it requires frequent matrix reconstruction and decomposition, increasing computational cost and limiting flexibility. In contrast, the state-space method enables flexible numerical integration and better captures fast transients and multi-timescale dynamics; it underpins tools such as MATLAB/Simulink, RT-LAB, PLECS, and has been adopted in RTDS, a representative nodal analysis-based real-time simulation tool, through the Universal Converter Model (UCM) to achieve more accurate converter simulations [3].

This work was supported by National Key R&D Program of China under Grant 2023YFC3807000, and National Natural Science Foundation of China under Grant 52377118.

Modeling approaches for power converters are typically classified into switch-based and external characteristic-based methods. The binary resistance switch model captures high-frequency switching transients but require frequent matrix updates, leading to high computational cost; techniques like the associated discrete circuit (ADC) model mitigate this by keeping system matrices constant, though at the expense of accuracy and parameter sensitivity [4]. In contrast, external characteristic-based models, such as switching function and average value models, offer higher efficiency but sacrifice generality and accuracy to some degree. For high-frequency switching converters, detailed switch-level models combined with pre-computed matrices and non-iterative prediction methods remain preferred in system-level EMT simulations.

To address the efficiency bottleneck caused by time-varying system topologies, algorithm-level optimizations have also been proposed. One common approach is circuit decoupling, which decouples the system into multiple subcircuits to reduce the number of switching states within each subsystem. Circuit decoupling is conventionally achieved by transmission line propagation delay, or delay-free methods such as compensation techniques [5] and the state-space nodal (SSN) method [6]. In addition, further improvement can be achieved at the numerical integration level. By splitting the computational matrix into constant and time-varying components, the computational burden caused by topology switching is reduced, achieving a balance between simulation accuracy and efficiency [7]. A similar approach is employed in [8] to decompose the matrix and obtain constant components. The delay-free decoupling methods generally provide better accuracy and numerical stability, although their algorithmic implementation is relatively more complex and thus typically suited to CPU-based simulation platforms.

From a hardware architecture perspective, although CPUs serve as the main computation units in most commercial real-time simulators (e.g., RTDS, RT-LAB, and Speedgoat systems), their sequential execution and limited parallelism create performance bottlenecks in simulations of converter-rich systems, such as microgrids. In contrast, FPGAs provide massive parallelism, distributed memory, and pipelined structure, enabling small time-step, high-fidelity EMT simulations and attracting increasing research attention [9]. RTDS integrates dedicated FPGAs to accelerate the simulation of high-frequency power electronic converters. RT-LAB also incorporates FPGAs to provide ultra-fast and low-latency processing for power electronic and high-speed control applications.

However, the hardware resources of FPGAs are inherently limited. Converter-based systems with numerous switching devices and pre-computed matrices impose heavy demands on FPGA logic and memory resources. Efficient FPGA-based real-time EMT simulation thus requires optimized utilization of spatial and temporal parallelism to balance computational efficiency, accuracy, and resource constraints. To this end,

979-8-3315-8138-1

this paper proposes an FPGA-based real-time simulation method for microgrids, achieving a trade-off among real-time performance, numerical precision, and hardware efficiency. The main contributions are as follows: 1) A delay-free decoupled real-time simulation method based on state-space splitting is proposed for microgrids, eliminating the need for network-level decoupling and associated delays or errors. 2) An FPGA-oriented solver architecture compatible with state-space splitting and exponential integrator is developed, achieving efficient parallel computation through solver-step mapping and pipelined scheduling. 3) Simulation studies are conducted to validate the proposed method, demonstrating its accuracy and hardware resource efficiency.

II. State-Space Splitting for Real-Time Simulation

The real-time simulation process under the state-space framework is closely related to the choice of numerical integration method. To effectively support the real-time EMT simulation of microgrids, the selected state-space integration method needs to balance computational time, memory requirements, and numerical accuracy. Exponential integrator introduces exponential operators in the solution of differential equations, provides excellent numerical accuracy and stability [10]. Moreover, it is inherently compatible with matrix splitting technique, effectively mitigating the computational and storage burden caused by frequent variations in the system state matrix.

A. Exponential Integrator

A multi-converter microgrid can be mathematically represented by a set of piecewise-linear state-space equations as follows:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}_i \boldsymbol{x}(t) + \boldsymbol{B}_i \boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}_i \boldsymbol{x}(t) + \boldsymbol{D}_i \boldsymbol{u}(t) \end{cases} \tag{1}$$

where, $\boldsymbol{x}(t)$ represents the state variables, $\boldsymbol{u}(t)$ represents the input variables, and $\boldsymbol{y}(t)$ represents the system outputs or measured quantities. $\boldsymbol{A}_i, \boldsymbol{B}_i, \boldsymbol{C}_i, \boldsymbol{D}_i$ are the time-varying system matrices corresponding to a specific power converter switching state i, where the combination number can be huge. The system exhibits the following state transition in the time period $[t_k, t_{k+1}]$, where the switching state stays unchanged as i. The change of switching state and corresponding system matrices are subject to the turning on/off criterion of each power electronic switches.

$$\boldsymbol{x}(t_{k+1}) = \mathrm{e}^{(t_{k+1}-t_k)\boldsymbol{A}_i} \boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathrm{e}^{(t_{k+1}-t)\boldsymbol{A}_i} \boldsymbol{B}_i \boldsymbol{u}(t) \mathrm{d}t \tag{2}$$

The above analytical expression involves the computation of matrix exponentials and convolution terms, where accurately evaluating the latter is computationally demanding. In EMT simulations, the piecewise-linear state equation in (1) typically features input variables that are DC or power-frequency AC sources. By introducing auxiliary state variables, the state equations can be transformed into an augmented autonomous system as shown in (3), thereby eliminating the convolution term in the state transition relation and simplifying the calculation.

$$\begin{cases} \dot{\widetilde{\boldsymbol{x}}}(t) = \widetilde{\boldsymbol{A}}_i \widetilde{\boldsymbol{x}}(t) \\ \boldsymbol{y}(t) = \widetilde{\boldsymbol{C}}_i \widetilde{\boldsymbol{x}}(t) \end{cases} \tag{3}$$

where $\widetilde{\boldsymbol{x}} = [\boldsymbol{x}, \boldsymbol{x}_u]^{\mathrm{T}}$ denotes the augmented state variables, and $\widetilde{\boldsymbol{A}}_i, \widetilde{\boldsymbol{C}}_i$ represent the augmented system matrices. Within the state-space framework of EMT simulations, numerical integration methods can be flexibly selected. The exponential integrator achieves the same A-stability as the trapezoidal method while maintaining explicit structure. The key of this

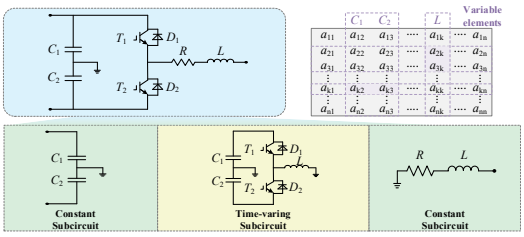


Fig. 1 Illustration of state-space matrix splitting.

method is to obtain the state transition matrix through the matrix exponential operator. For the state-space equations in (3), the single-step exponential integration formula can be expressed as:

$$\begin{cases} \widetilde{\boldsymbol{x}}(t_{n+1}) = \mathrm{e}^{h\widetilde{\boldsymbol{A}}_i} \widetilde{\boldsymbol{x}}(t_n) \\ \boldsymbol{y}(t_{n+1}) = \widetilde{\boldsymbol{C}}_i \widetilde{\boldsymbol{x}}(t_{n+1}) \end{cases} \tag{4}$$

where h represents the simulation step-size. The system dynamic process can be explicitly solved through this equation.

B. State-Space Splitting

In the state-space framework, $\widetilde{\boldsymbol{A}}_i$ characterizes the coupling relationships among state variables. When a switching event occurs, the coupling between state variables on both sides of the switch changes, leading to corresponding variations in specific matrix elements. For microgrids, each switch typically affects only a small number of elements, primarily located in the rows and columns associated with switch adjacent components. According to whether the matrix elements vary with the switching state, the system state matrix can be split into a constant part and a time-varying part. The identification of time-varying elements is illustrated in Fig. 1.

$$\widetilde{\boldsymbol{A}}_i = \boldsymbol{M} + \boldsymbol{N}_i \tag{5}$$

where $\boldsymbol{M}$ denotes the constant matrix and $\boldsymbol{N}_i$ represents the time-varying matrix. This state-space splitting method, based on time-varying characteristics, essentially achieves an energy-decoupled representation of microgrids. The energy exchange processes of storage components (inductors and capacitors) directly connected to the converters are separated, appearing respectively in the time-varying subcircuits related to switching states and the constant subcircuits.

For the matrix exponential term $\mathrm{e}^{h\widetilde{\boldsymbol{A}}_i} = \mathrm{e}^{h(\boldsymbol{M}+\boldsymbol{N}_i)}$, according to the Suzuki–Trotter decomposition principle for matrix exponentials [11], it can be expressed as the decomposition shown in (6), thereby dividing the overall computation into two stages. Since $\boldsymbol{M}$ is a constant matrix, the exponential term $\mathrm{e}^{h\boldsymbol{M}}$, can be accurately computed offline and the results can be pre-stored in the FPGAs. This strategy effectively reduces online computation time and lowers the real-time solving complexity during simulation.

$$\mathrm{e}^{h\widetilde{\boldsymbol{A}}_i} = \mathrm{e}^{h(\boldsymbol{M}+\boldsymbol{N}_i)} = \mathrm{e}^{h\boldsymbol{M}} \mathrm{e}^{h\boldsymbol{N}_i} + O(h^2) \tag{6}$$

Since the time-varying matrix $\boldsymbol{N}_i$ is inherently sparse and decoupled, the matrix exponential $\mathrm{e}^{h\boldsymbol{N}_i}$ is low in computation burden. To strictly match the error order shown in (6) and improve computational efficiency in real-time simulation, the exponential term $\mathrm{e}^{h\boldsymbol{N}_i}$ can be further replaced with its low-order approximation, as shown in (7). Inevitably, this approximation introduces an error term $h^2\boldsymbol{N}_i^2/2$. However, unlike fully explicit forward Euler method, the system dynamics governed by the constant matrix retain the exact matrix exponential form, which restricts potential numerical instability issues to the local branches represented by the time-varying matrix. Furthermore, benefiting from the small time-step enabled by the FPGA parallel mechanism, a favorable

trade-off between computational efficiency and numerical accuracy can be achieved in real-time simulation.

$$e^{hN_i} = \boldsymbol{I} + h\boldsymbol{N}_i + h^2\boldsymbol{N}_i^2/2 + O(h^3) \approx \boldsymbol{I} + h\boldsymbol{N}_i \quad (7)$$

After the aforementioned state-space splitting and numerical approximation, the FPGA based real-time simulation process can be expressed as:

$$\begin{cases} \tilde{\boldsymbol{x}}(t_{n+1}) = e^{h\boldsymbol{M}}(\boldsymbol{I} + h\boldsymbol{N}_i)\tilde{\boldsymbol{x}}(t_n) \\ \boldsymbol{y}(t_{n+1}) = \tilde{\boldsymbol{C}}_i\tilde{\boldsymbol{x}}(t_{n+1}) \end{cases} \quad (8)$$

The approximation error in this numerical scheme can be analyzed by inspecting $\boldsymbol{E}_s = e^{h(\boldsymbol{M}+\boldsymbol{N}_i)} - e^{h\boldsymbol{M}}(\boldsymbol{I} + h\boldsymbol{N}_i)$, which reveals that the error mainly consists of the operator splitting error and the approximate error of the time-varying matrix exponential. The estimated error boundary is shown as:

$$\begin{aligned} \|\boldsymbol{E}_s\tilde{\boldsymbol{x}}_n\| &\approx \|(\boldsymbol{N}_i\boldsymbol{M} - \boldsymbol{M}\boldsymbol{N}_i + \boldsymbol{N}_i\boldsymbol{N}_i)\cdot\tilde{\boldsymbol{x}}_n\|\cdot h^2/2 \\ &\le (2\|\boldsymbol{M}\|\|\boldsymbol{N}_i\| + \|\boldsymbol{N}_i\|^2)\|\tilde{\boldsymbol{x}}_n\|\cdot h^2/2 \end{aligned} \quad (9)$$

As the system scale expands, the errors are still mainly localized in regions associated with switching devices. In the real-time simulation with small time-step, while meeting the requirements of numerical stability, the accumulated errors are observed to remain bounded and acceptable.

Within the state-space framework, employing matrix splitting through exponential integrator effectively reduces the computational complexity, ensures controllable error with verifiable numerical stability, and more importantly, significantly alleviates matrix pre-storage demands.

III. FPGA Implementation of State-Space Splitting

The computation in the state-space method is based on the system's state-space equations. In real-time simulation, the calculation process does not directly involve individual electrical components. After the host computer processes the system topology and component parameters, the data transmitted to the FPGAs only contains numerous matrices. This design endows state-space based real-time simulation with high universality and compatibility. Moreover, FPGAs are well suited for computational tasks characterized by simple flows and large-scale operations. Since the state-space method primarily involves matrix operations with numerical parallelism, it can fully exploit the advantages of FPGAs.

A. Overall Hardware Architecture

Based on state-space method, this paper proposes an FPGA-based real-time EMT simulation framework for microgrids, developed using the state-space splitting method with exponential integrator. By partitioning the computation process, the proposed approach achieves delay-free decoupling at the algorithmic level and efficient mapping at the hardware level. The overall framework consists of such modules: global control module, power system solver, control system solver, and a data exchange module, shown in Fig. 2.

The global control module is responsible for managing the overall simulation process, including generating the start and reset signals for each simulation time step, computing the actual time, and producing sampling signals. Within each simulation step, the power system solver and control system solver execute their computational tasks in parallel. Upon completion, data are exchanged between them through the data exchange module to ensure consistency.

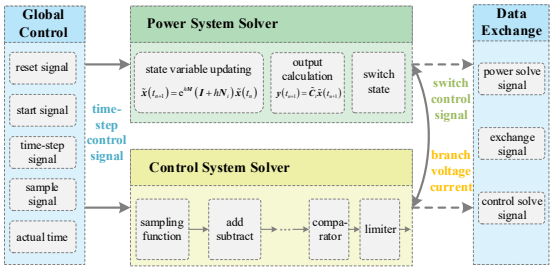


Fig. 2 The overall hardware architecture.

The control system typically consists of multiple control components described by input-output relationships, featuring a relatively sparse structure with only a few feedback loops. In this work, the control system solver primarily performs sequential solving based on the input-output relationships of each control loop. For feedback elements, a one-step delay is inserted to decouple them, converting them into explicit calculations. It is important to note that this specific delay is introduced solely for the control feedback loop and is distinct from the delay-free decoupling method applied to the power network solution. Furthermore, such treatment for feedback loops is widely adopted in commercial real-time simulators to ensure deterministic computation time.

B. Power System Calculation Step Mapping

The power system solver performs real-time computation based on the time-stepping formula given in (8). The main computational procedures include matrix updating, state variable updating, and output calculation. The proposed power system solver module is specifically shown in Fig. 3. Its structure corresponds to the mapping relationship of the state-space splitting solution formula on the hardware, thereby achieving the collaborative design of algorithm and hardware architecture.

The specific solver modules are described as follows. In the splitting formulation, $e^{h\boldsymbol{M}}$ represents the dense matrix corresponding to the constant linear part of the system. This matrix can be pre-stored in the FPGAs as a constant coefficient matrix. High-throughput dense matrix-vector calculation can be achieved through a parallel multiply-accumulate array organized in a pipelined structure. Meanwhile, to balance real-time performance and hardware resource constraints, a row-block reuse strategy is introduced in the hardware design, which further reduces on-chip memory usage while maintaining computational throughput.

In contrast, $h\boldsymbol{N}_i$ is a sparse matrix that reflects the time-varying characteristics. In microgrids, the number of matrix elements affected by the switching state of each power electronic device is limited. Typically, these states only correlate with the state variables at the device ports, and the influence regions of different devices remain mutually independent. Based on this property, the time-varying matrix is partitioned into several independent sub-blocks according to device boundaries. Each sub-block is stored using a block compressed storage structure, which effectively reduces both the memory footprint and access overhead.

During the system solution process, the computational time of the designed sparse matrix-vector multiplication module is linearly proportional to the number of nonzero elements in the matrix. Prioritizing the calculation of $(\boldsymbol{I} + h\boldsymbol{N}_i)$ would partially destroy the matrix sparsity and reduce parallel efficiency. Therefore, the sparse matrix solution workflow was adjusted to $\left(\tilde{\boldsymbol{x}}(t_n) + h\boldsymbol{N}_i\tilde{\boldsymbol{x}}(t_n)\right)$. This modification does not affect the overall solution accuracy but maximizes the utilization of both the matrix sparsity and the parallel computing characteristics of the FPGAs.

Furthermore, since the splitting formula divides the computation into two numerical integration steps with a strict sequential relationship, this paper further incorporates an operation module reuse mechanism in the design. Although

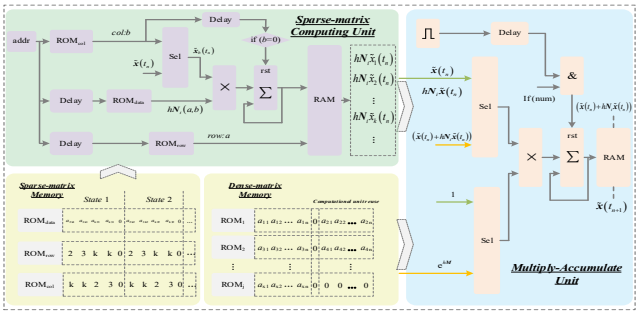


Fig. 3 Power system solver module structure.

vector multiplication operations for dense and sparse matrices exhibit different characteristics and are not suitable for sharing hardware units, the vector summation components in both integration steps share consistent forms and operate independently. Consequently, the same set of solution modules can be reused to further reduce hardware resource consumption.

The output computation also exhibits sparsity, but the matrix lacks all-zero rows, making it more suitable for the traditional compressed sparse row format [12]. The specific solution is performed by a set of multiply-accumulate units in a pipelined structure.

C. Switch State Prediction

Based on the matrix computation module design, this paper further designs a switch status prediction and synchronization processing for the characteristic of frequent topology switching in microgrids. This module employs a pipeline structure to sequentially process different switch types and their corresponding decision criteria. Simultaneously, synchronous switching prediction is introduced during switch state processing. Inductive currents at power electronics device ports serve as additional measurement variables to predict potential synchronous switching behaviors in the next time-step [13]. Corresponding switch signals are adjusted in advance to avoid unreasonable arm on-off states.

Through analysis of the state-space splitting process, this paper achieves a complete mapping and optimization loop from the algorithm layer to the FPGAs hardware layer. It fully leverages hardware resources to realize parallelization of the computational process and module reuse.

IV. Case Study

In this section, the real-time performance, accuracy, and resource optimization characteristics of the proposed method are validated through test cases. The proposed method is implemented on an Altera Stratix® V 5SGSMD5K2F40C2 FPGA development board as the computation hardware. The FPGA features 172600 ALMs, 44.27 Mb of on-chip memory, and 1590 DSP blocks.

A. Grid-Connected PV System

A grid-connected PV system is employed to benchmark the performance of the proposed framework against the common Nodal-Analysis FPGA real-time simulation approach. The system topology is illustrated in Fig. 4, where the PV unit is connected to the grid at $t = 0.06\ s$.

Table I presents the comparison of hardware resource consumption, calculation time of each module, and relative error between the two simulation frameworks. In the Nodal-Analysis FPGA simulator, the switches are modeled using the

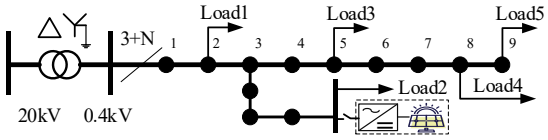


Fig. 4 Grid-connected PV system.

ADC model. As indicated by the data, the proposed FPGA real-time simulator based on State-Space Splitting demonstrates superior potential in all aspects.

TABLE I. Comparison of Simulation Results

		State-Space	Nodal-Analysis
Hardware Resource	**Logic Resources** (172,600)	51.63%	89.24%
	Memory Bits (41,246,720)	11.98%	20.14%
	DSP Blocks (1590)	14.97%	25.79%
Calculating Time	**Power System**	1.392 μs	2.328 μs
	Control System	1.864 μs	1.864 μs
	Data Exchange	0.096 μs	0.096 μs
	Total Time	1.960 μs	2.424 μs
Mean Relative Error		0.0714%	0.2143%

A comparison is conducted between the FPGA real-time simulation and the EMTP-RV offline simulation. Fig. 5 depicts the voltage waveforms on the DC side of the inverter. The results reveal that the proposed framework achieves high consistency with the offline results, exhibiting a lower error magnitude compared to the ADC model within the Nodal-Analysis framework.

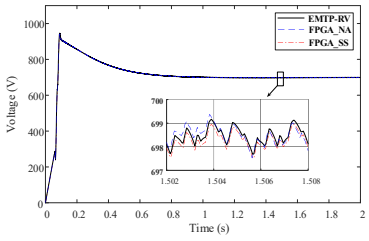


Fig. 5 DC side voltage of grid-connected PV system.

B. DC Microgrid

The effectiveness of the proposed framework is further verified through a DC microgrid case incorporating multiple converters. The system topology is shown in Fig. 6. In the test case, multiple devices are connected to the DC bus via converters, while the DC bus voltage is mainly maintained by the grid-connected converter and the energy storage unit on the left-side. The test case contains 7 power converters and 22 detail modeled power electronic switches.

The simulation scenario is that a 20% reduction in DC load occurs at $t = 2\ s$, with the simulation time step set to 3 μs. Meanwhile, the offline simulation results obtained from EMTP-RV are used as a reference. Fig. 7(a)-(d) depict the DC bus voltage and key branch currents, and Table II reports the relative errors. Notably, the relative errors remain at a low

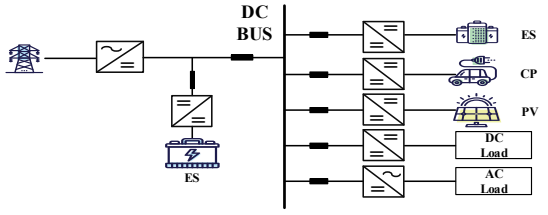
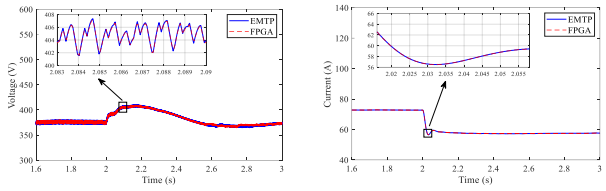
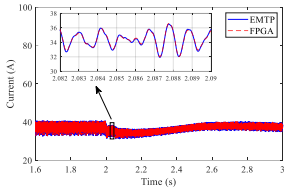


Fig. 6 DC microgrid case.

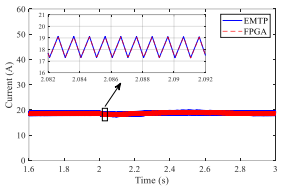


(a) DC bus voltage

(b) DC load current



(c) PV grid-connected line current



(d) Energy storage converter inductor current

Fig. 7 Simulation results of DC microgrid test case.

level, exhibiting close agreement with the EMTP-RV reference, thereby verifying the correctness of the proposed state-space splitting framework. The remaining discrepencies primarily stem from differences in the solution formulation, numerical integration schemes, and control system solutions; nevertheless, the overall error remains small.

TABLE II. RELATIVE ERROR OF SIMULATION RESULTS

Relative Error	DC bus voltage	DC load current	PV line current	ESC current
Max	0.9350%	0.3087%	2.3832%	0.7752%
Mean	0.1650%	0.0393%	0.4418%	0.1400%

In addition, the summary of several key indicators in terms of hardware resource consumption are shown in Table III.

TABLE III. HARDWARE RESOURCE CONSUMPTION

	Logic Resources (172,600)	Memory Bits (41,246,720)		DSP Blocks (1590)
Occupy	62.20%	10.64%		15.60%
		Dense matrix	**Sparse matrix**	
		10.30%	0.13%	

Based on the results in the Table, it can be seen that by using the state-space splitting method proposed in this paper, the hardware resources occupied by matrix storage are significantly reduced. Sparse matrix storage accounts for only 0.13% of the total resources. This improvement is attributed to the splitting method proposed in this paper and the corresponding sparse matrix storage and numerical approximation algorithm, despite the many power converters in microgrids and therefore the large number of power electronic switch state combinations. Without matrix splitting, the number of dense matrices would increase exponentially with the number of switching devices, where a $2^{22} \approx 4M$ fold increase in dense matrix storage is expected which leads to excessive memory usage that could exceed FPGA resource limits. Furthermore, since the matrix dimension of the microgrids in the state-space framework is generally lower than that in the nodal framework, the hardware resource consumption of the Nodal-Analysis simulation exceeded the available capacity of the FPGA, rendering it impossible to implement for this test case. The case study verifies the correctness and resource efficiency of the proposed method and algorithm–hardware co-design, effectively addressing the challenges of real-time simulation for microgrids.

V. CONCLUSIONS

This study proposes a real-time EMT simulation method for microgrids based on state-space splitting using FPGAs. By employing exponential integrator and matrix splitting, the system matrix achieves delay-free decoupling, significantly reducing computational and storage demands while ensuring numerical accuracy and stability. Combined with the parallel architecture of FPGAs, a hardware mapping and pipeline computing structure for matrix splitting solution is designed. The simulation results verify the accuracy and resource efficiency characteristics of this method in high-frequency switching systems, providing new ideas and implementation approaches for the efficient real-time simulation of microgrids.

REFERENCES

[1] H. Fu, P. Li, X. Fu, Z. Wang, J. Yan, J. Wu, and C. Wang, "Asynchronous multi-rate method of real-time simulation for active distribution networks," *IET Renew. Power Gener.*, vol. 17, no. 1, pp. 150–161, Jan. 2023.

[2] J. Xu, P. Wu, Z. R. Li, K. Y. Wang, G. J. Li, and B. Han, "Switching-period-synchronization-based real-time simulation method suitable for power converters with high switching frequency," *IEEE Trans. Ind. Electron.*, vol. 72, no. 10, pp. 10215–10226, Oct. 2025.

[3] Y. Qi, H. Ding, S. Shi, Y. Zhang, and A. Gole, "High efficiency modeling of multi-layer cascaded dual-active-bridge (DAB) units on real-time simulator," in *2024 IEEE Power & Energy Society General Meeting (PESGM)*, Seattle, WA, USA: IEEE, July 2024, pp. 1–5.

[4] Q. Mu, J. Liang, X. Zhou, Y. Li, and X. Zhang, "Improved ADC model of voltage-source converters in DC grids," *IEEE Trans. Power Electron.*, vol. 29, no. 11, pp. 5738–5748, Nov. 2014.

[5] B. Bruned, J. Mahseredjian, S. Dennetière, J. Michel, M. Schudel, and N. Bracikowski, "Compensation method for parallel and iterative real-time simulation of electromagnetic transients," *IEEE Trans. Power Deliv.*, vol. 38, no. 4, pp. 2302–2310, Aug. 2023.

[6] C. Dufour, J. Mahseredjian, and J. Belanger, "A combined state-space nodal method for the simulation of power system transients," *IEEE Trans. Power Deliv.*, vol. 26, no. 2, pp. 928–935, Apr. 2011.

[7] X. Fu, W. Wu, P. Li, J. Mahseredjian, J. Wu, and C. Wang, "Splitting state-space method for converter-integrated power systems EMT simulations," *IEEE Trans. Power Deliv.*, vol. 40, no. 1, pp. 584–595, Feb. 2025.

[8] M. Xu, W. Gu, Y. Cao, S. Chen, F. Zhang, and W. Liu, "A state variables elimination-based EMTP-type constant admittance equivalent modeling method for power electronic converters," *IEEE Trans. Power Deliv.*, vol. 40, no. 2, pp. 1100–1113, Apr. 2025.

[9] H. Fu, P. Li, X. Fu, J. Yan, Z. Wang, K. Wang, J. Wu, and C. Wang, "A compact real-time simulator with spatial-temporal parallel design for large-scale wind farms," *CSEE J. Power Energy Syst.*, vol. 9, no. 1, pp. 50–65, 2023.

[10] C. Wang, X. Fu, P. Li, J. Wu, and L. Wang, "Multiscale simulation of power system transients based on the matrix exponential function," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 1913–1926, May 2017.

[11] M. Suzuki, "Convergence of general decompositions of exponential operators," *Commun. Math. Phys.*, vol. 163, no. 3, pp. 491–508, July 1994.

[12] Z. Koza, M. Matyka, S. Szkoda, and L. Miroslaw, "Compressed multirow storage format for sparse matrices on graphics processing units," *SIAM Journal on Scientific Computing*, vol. 36, no. 2. pp. C219–C239, 2014.

[13] T. Maguire, S. Elimban, E. Tara and Y. Zhang, "Predicting Switch ON/OFF Statuses in Real Time Electromagnetic Transients Simulations with Voltage Source Converters," *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, Beijing, China, 2018, pp. 1–7.