

Parallel-in-Time Power System Simulation Using a Differential Transformation Based Adaptive Parareal Method

YANG LIU¹ (Member, IEEE), BYUNGKWON PARK² (Member, IEEE),
KAI SUN¹ (Senior Member, IEEE), ALEKSANDAR DIMITROVSKI³ (Senior Member, IEEE),
AND SRDJAN SIMUNOVIC⁴

¹Department of EECS, The University of Tennessee Knoxville, Knoxville, TN 37996 USA

²Department of Electrical Engineering, Soongsil University, Seoul 06978, South Korea

³Department of ECE, University of Central Florida, Orlando, FL 32816 USA

⁴Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830 USA

CORRESPONDING AUTHOR: K. SUN (kaisun@utk.edu)

This work was supported by UT-Battelle, LLC, through the U.S. Department of Energy, under Contract DE-AC05-00OR22725.

ABSTRACT For parallel-in-time simulation of large-scale power systems, this paper proposes a differential transformation based adaptive Parareal method for significantly improved convergence and time performance compared to a traditional Parareal method, which iterates a sequential, numerical coarse solution over extended time steps to connect parallel fine solutions within respective time steps. The new method employs the differential transformation to derive a semi-analytical coarse solution of power system differential-algebraic equations, by which the order and time step, as well as the window length with a multi-window solution strategy, can adaptively vary with the response of the system. Thus, the new method can reduce divergences and also speed up the overall simulation. Extensive tests on the IEEE 39-bus system and the Polish 2383-bus system have verified the performance of the proposed method.

INDEX TERMS Differential transformation, Parareal, parallel algorithms, power system simulation, power system stability.

I. INTRODUCTION

RENEWABLE intermittent energy resources keep penetrating electrical power systems and bring more uncertainties and challenges to power system operations [1]. Specifically, a day-ahead power system dynamic security assessment for representative or predicted operating conditions is no longer adequate for an interconnected power grid with a high penetration of renewable energy resources because its generation is far less predictable than before. Thus, dynamic security assessment tools that possess a real-time or even faster-than-real-time computing capability will be highly valuable to accommodate more uncertain and diversified operating conditions [2].

Time-domain simulation has been widely used for dynamic security assessment, especially transient stability assessment, of a large-scale power grid under contingencies [3], [4]. Compared to direct methods for transient stability analysis, time-domain simulation has high accuracy and is capable of

dealing with a wide variety of detailed power system models. On the other hand, its major drawback lies in the huge computational burden and numerical divergence issues when solving many nonlinear differential-algebraic equations (DAEs) that model the power grid [5], [6]. Hence, solutions to boost the computational performance while ensuring the convergence of the solver algorithm in time-domain simulation to meet the real-time or faster-than-real-time requirements have been a challenging problem in power system research.

Recent studies on speeding up power system simulations fall into two categories: **1)** utilization of parallel computing via parallel-in-space methods such as the multi-decomposition method [7], domain decomposition method in [8] and [9], and multi-area Thévenin equivalents in [10], or parallel-in-time methods such as the waveform relaxation method in [11] and Parareal method in [12], [13], [14], [15], [16], [17], [18], and [19]; and **2)** semi-analytical simulation methods such as the Adomian decomposition method in

[20] and [21], time-power series method in [22], differential transformation method in [23] and [24], and holomorphic embedding method in [25].

In the parallel computing category, existing methods mainly parallelize computations among multiple simulation runs for different contingencies. They still have limited ability of parallelization on a single simulation run, even with simplified models and many parallel processors in high-performance computers [26]. In this respect, the Parareal method has been successful in temporal parallelization of computations with low-dimensional models in other fields such as analysis of molecular dynamics and fluid-structure computation [27], [28]. The basic idea of the Parareal method is below. First, a computationally efficient coarse solver is employed to initialize the state variables for future time steps. Then, a fine solver is used to solve all coarse time steps in parallel, whose solutions are used to update the initial values of each coarse time step. Finally, the above procedure is performed iteratively until convergence. However, when applied to simulations of high-dimensional power system DAE models, a conventional Parareal method can face numerical divergence issues as discussed in [12], [13], [14], [15], [16], [17], [18], and [19].

For the methods in the semi-analytical simulation category, the derivation of a high-order semi-analytical solution (SAS) for high-dimensional power system DAEs can be challenging due to intensive symbolic computations of analytical solutions. Such challenges can be overcome by the recently proposed differential transformation (DT) method [23], [24], which allows a more efficient, recursive procedure for deriving an SAS from low order to high order. Thus, it allows one to use an SAS of any desired order and accuracy for the purpose of, e.g., an expected shorter or longer time step length in time-domain simulations. Considering these observations, power system simulation requires and benefits from further improvements, which innovatively integrate methods from the two categories to create a more adaptive and effective parallel-in-time algorithm for enabling the prediction of power system dynamic behaviors in real-time.

It is worth mentioning that an adaptive Parareal method [29] was recently proposed in the applied mathematics field and was tested on simple low-dimensional differential equations. However, it focuses on adjusting the accuracy of the fine solver to improve the computational efficiency, which is different from this paper that aims at developing the adaptive coarse solver and other adaptive strategies for improving both the convergence performance and computational efficiency in a high-dimensional nonlinear power system model. Currently, there is no mature method to overcome the divergence issues of the Parareal method in high-dimensional nonlinear power systems models.

In this regard, the contributions of this paper include: 1) a variable-order variable-step variable-window (VO-VS-VW) adaptive Parareal method is proposed for parallel-in-time power system simulations with greatly enhanced convergence and computation speed; 2) the DT method, for

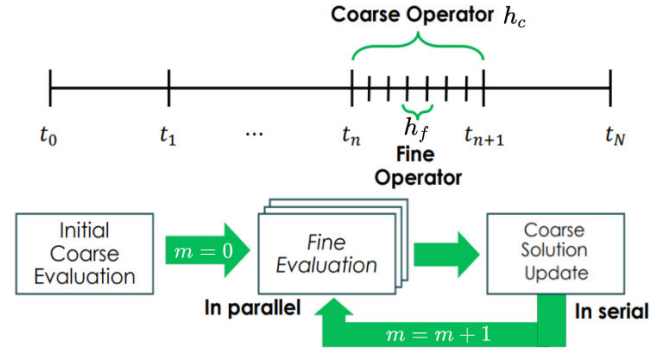


FIGURE 1. Illustration of the Parareal method.

the first time, is employed as the coarse solver in a Parareal method to enable its better adaptivity than a numerical coarse solver; 3) the performance of the proposed method is compared to a traditional Parareal method by extensive tests on a 39-bus system and a 2383-bus system.

In the rest of the paper, Section II introduces the traditional Parareal method using numerical-based coarse solvers with fixed multi-window and first-order coarse solution update strategies. Section III presents the proposed DT-based adaptive Parareal method, with case studies presented in Section IV, and conclusions drawn in Section V.

II. TRADITIONAL PARAREAL METHOD

Notation: The following notations are used in the rest of the paper:

- m, M index and the number of Parareal iterations
- k, K index and the order of DT
- n, N index and the number of time steps in the coarse solver
- w, c, f superscripts of Parareal window, coarse and fine step

Consider the power system model in the form of

$$\begin{cases} \dot{x}(t) = f(x(t), v(t)) \\ 0 = g(x(t), v(t)), \end{cases} \quad \begin{cases} x(0) = x_0 \\ v(0) = v_0 \end{cases} \quad (1)$$

where x is the vector of state variables, v is the vector of bus voltages, and f and g are functions in differential equations and algebraic equations, respectively.

For simplicity, let $\mathbf{x}(t) = [x(t)^T, v(t)^T]^T$, $\mathbf{x}_0 = [x_0^T, v_0^T]^T$ and rewrite it as

$$0 = \mathbf{f}(\mathbf{x}(t), \dot{\mathbf{x}}(t)), \mathbf{x}(0) = \mathbf{x}_0 \quad (2)$$

Fig. 1 illustrates the basic idea of the Parareal method for simulating a power system model. First, the desired simulation length $[t_0, t_N]$ is decomposed into N coarse steps with length h_c , and each coarse step is further decomposed into many fine steps with length h_f . Then, the block of **initial coarse evaluation** computes the initial coarse solutions $\mathbf{x}_n^{c,0}$ by performing the simulation over $[t_0, t_N]$ in serial, using a coarse operator \mathcal{C} with the coarse step length h_c . Afterward, the iteration process is performed between the blocks of fine

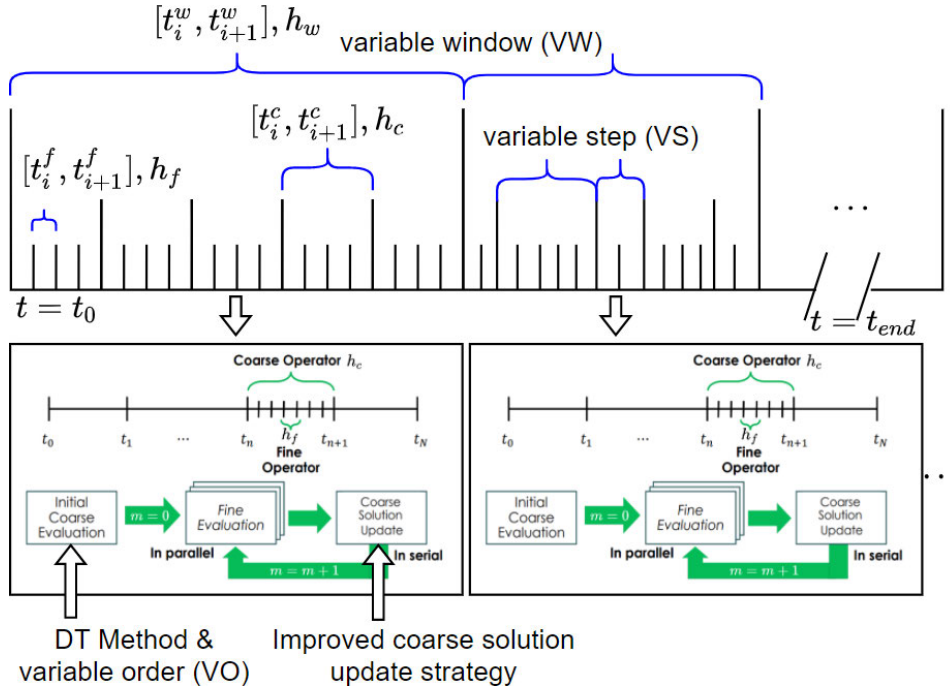


FIGURE 2. Illustration of the proposed method.

evaluation and the coarse solution update. The block of **fine evaluation** is to provide a better solution $\mathbf{x}_n^{f,0}$ at each interval by simulating each coarse interval $[t_n, t_{n+1}]$ in parallel using a fine operator \mathcal{F} with the fine step length h_f . The block of **coarse solution update** is to correct the coarse solution at each coarse interval. The iteration process continues until the stopping criteria is met, e.g., the differences of coarse solutions between the m^{th} iteration and the $(m+1)^{\text{th}}$ iteration are smaller than a pre-defined threshold, or the maximum number of iterations is reached.

The Parareal method described above, having been successful in other fields and having shown promising results for speeding up power system simulations, can face divergence issues when being applied to nonlinear high-dimensional power system models. The simulation length to obtain the fast and robust convergence of the Parareal method for a power system model is often restricted to around 1 second. When a longer simulation length is desired, e.g., 5 to 10 seconds in transient stability assessment, a multi-window strategy was utilized in [12], [13], [14], [15], [16], [17], [18], and [19] which decomposes the desired simulation length $[t_0, t_N]$ into multiple fixed windows with a window length of h_w . Then, the Parareal method is applied to each window sequentially, and the trajectories of state variables in all windows are connected together to give the trajectory of desired simulation length.

III. PROPOSED DT-BASED ADAPTIVE PARAREAL METHOD

A. OVERVIEW OF THE PROPOSED METHOD

Fig. 2 illustrates the basic idea of the proposed method. The following improvements are made compared with the

conventional Parareal method in Fig. 1. First, the DT method is employed as the coarse solver to replace a traditional numerical integration method as detailed in Section III-B. Second, a variable-order variable-step DT method is further designed to utilize the adaptivity of the DT method to adjust the accuracy of the coarse solver adaptively, as is shown in Section III-C-1). Third, the Parareal method is applied to multiple time windows and a variable-window strategy is designed to allow further adaptivity of the algorithm, with details presented in Section III-C-2). Finally, an improved coarse solution update strategy is designed as discussed in Section III-C-3).

B. DT-BASED PARAREAL METHOD

The selection of the coarse operator plays a significant role in the convergence performance and computational efficiency of the Parareal method. The coarse operator needs to have sufficient accuracy to ensure the convergence of the Parareal method on one hand and needs to be fast enough to achieve considerable speedup on the other hand. The latter is due to the coarse operator evaluation in serial, which can offset the efficiency gain from the parallel computing and should be minimized. Therefore, selecting a proper coarse solver with sufficient accuracy and efficiency would be key to improving the convergence performance and time performance of the Parareal method. For this purpose, this section proposes a DT-based Parareal method that replaces the coarse solver in the traditional Parareal method by a DT solver. Our previous work has demonstrated that the DT method achieves better accuracy and efficiency than various conventional numerical integration methods. Moreover, the coarse solutions obtained

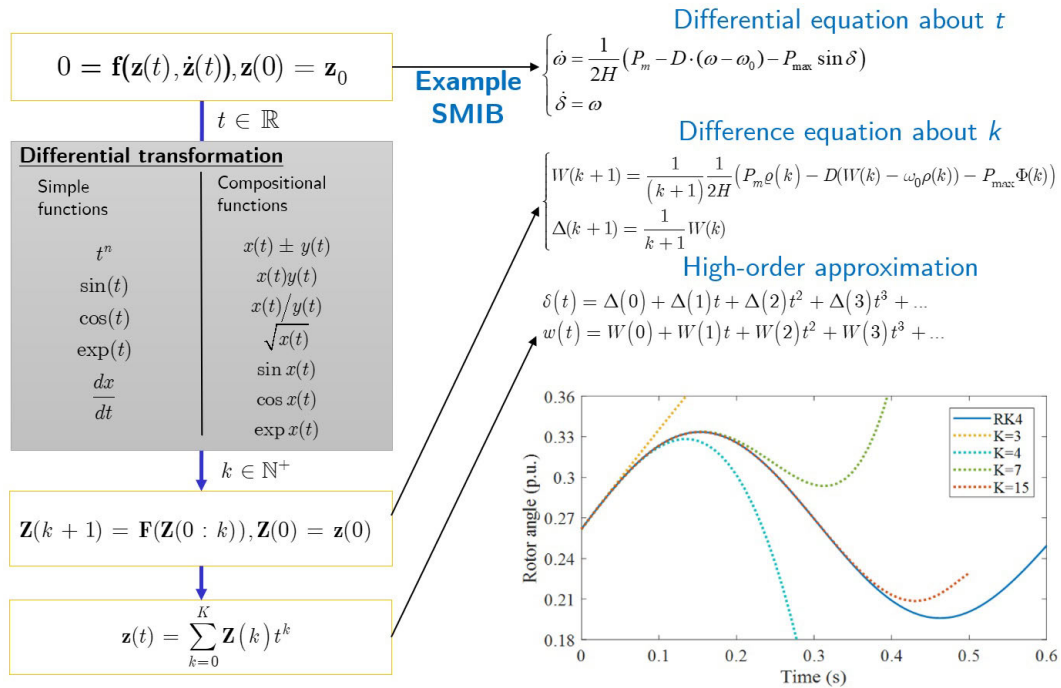


FIGURE 3. Illustration of the DT method.

by the DT method are in the form of power series of time, which further allow adaptively adjusting the order and time step length to balance between the accuracy and computation time in evaluating the coarse solutions, thus improving the overall performance of the Parareal method.

The motivation, definition, and transformation rules of the DT method were described in [23] and [24] and are briefly summarized in the Appendix. In short, the DT method provides various transformation rules for numerous generic functions such that a differential equation in a continuous set about the variable t (time) is converted to a new set of difference equations in a discrete set about the variable k (the power series order). The transformation rules for widely used linear and nonlinear functions in power system models are provided in [18] and [19].

Fig. 3 illustrates the idea of deploying the DT method in deriving semi-analytical solutions of differential equations or differential-algebraic equations in the form of high-order power series of time.

The left-hand side of Fig. 3 gives the procedure of applying the DT method. First, the original differential equations of power system models are transformed to a new set of difference equations by the DT method. The derivation of the difference equation is performed offline and is a one-time effort. Then, based on the obtained difference equation, the power series coefficients up to any desired order could be efficiently computed recursively. Finally, the solutions are approximated by summing all the calculated power series terms.

The right-hand side of Fig. 3 uses the swing equation of a single-machine-infinite-bus (SMIB) system to illustrate the derivation procedure. Then, the rotor angle trajectories obtained from the power series solution with different orders are compared with the benchmark solution obtained by the RK4 method with a small enough time step length for accuracy. It shows the convergence region of the DT method increases with the order of the power series solution. Especially, the solution with an order of 15 is accurate enough up to 0.35 seconds, which is more than half period of the rotor angle trajectory.

Algorithm 1 shows the detailed procedure of the DT method where: the offline stage derives a recursive equation about power series coefficients; line 1 in the online stage is the initialization of variables; lines 3 to 5 are the online evaluation of power series coefficients; line 6 is to compute $\mathbf{x}(t_{n+1})$ by summing the power series terms from 0th to K^{th} orders; line 7 moves the simulation to the next time step.

Algorithm 2 shows the detailed procedure of the DT-based Parareal method where: lines 1 and 2 correspond to the block of initial coarse evaluation in Fig. 2 and \mathcal{C}_{DT} is the DT-based coarse solver in Algorithm 1; lines 4 to 6 correspond to the block of fine evaluation; lines 7 to 10 correspond to the coarse solution update; lines 11 to 13 are the stopping criterion. In the algorithm, $\mathbf{x}_n^{c,m}$, $\mathbf{x}_n^{f,m}$, $\mathbf{x}_n^{*,m}$ respectively mean the coarse solution, the fine solution, and the corrected coarse solution at time instant t_n in the m^{th} iteration; tol is a pre-defined threshold; M is the recorded number of iterations for the Parareal method to converge; m_{max} is the pre-defined

Algorithm 1 DT Method

input: $t_0, t_{end}, \mathbf{x}_0, h, K$
output: $\mathbf{x}(t_n), t_0 \leq t_n \leq t_{end}$
offline stage: derive the recursive equation $\mathbf{X}(k+1) = \mathbf{F}(\mathbf{X}(0:k))$
online stage:

1. $\mathbf{x}(t_0) \leftarrow \mathbf{x}_0, t \leftarrow t_0, \mathbf{X}(0) \leftarrow \mathbf{x}_0, n \leftarrow 0$
2. **while** $t+h < t_{end}$
3. **for** $k = 0 : K$ // computer power series coefficients
4. $\mathbf{X}(k+1) = \mathbf{F}(\mathbf{X}(0:k));$
5. **end**
6. $\mathbf{x}(t_{n+1}) \leftarrow \sum_{k \in \{0,1,\dots,K\}} \mathbf{X}(k) h^k$
7. $t \leftarrow t+h, n \leftarrow n+1, \mathbf{X}(0) \leftarrow \mathbf{x}(t_{n+1})$
8. **end**

Algorithm 2 DT-Based Parareal Method

input: $t_0, t_N, h_c, h_f, \mathbf{x}_0$
output: $\mathbf{x}_n, n = 1, 2, \dots, N; M$

1. $\mathbf{x}_0^{*,0} = \mathbf{x}_0, \mathbf{x}_0^{*,1} = \mathbf{x}_0^{*,0}$
2. $\mathbf{x}_n^{*,0} = \mathbf{x}_n^{c,0} \leftarrow \mathcal{C}_{DT}(t_0, t_n, h_c, \mathbf{x}_0), n = 1, 2, \dots, N$
// initial coarse evaluation
3. **for** $m = 1 : m_{\max}$
4. **for** $n = m : N$ // fine evaluation in parallel
5. $\mathbf{x}_n^{f,m} \leftarrow \mathcal{F}(t_n, t_{n+1}, h_f, \mathbf{x}_{n-1}^{*,m-1})$
6. **end**
7. **for** $n = m : N$ // coarse solution update in serial
8. $\mathbf{x}_n^{c,m} \leftarrow \mathcal{C}_{DT}(t_{n-1}, t_n, h_c, \mathbf{x}_{n-1}^{*,m})$
9. $\mathbf{x}_n^{*,m} \leftarrow \mathbf{x}_n^{c,m} + \mathbf{x}_n^{f,m} - \mathbf{x}_n^{c,m-1}$
10. **end**
11. **if** $\|\mathbf{x}_n^{*,m} - \mathbf{x}_n^{*,m-1}\|_1 < tol, \forall n = 1, 2, \dots, N;$
// stopping criterion
12. $\mathbf{x}_n \leftarrow \mathbf{x}_n^{*,m}, n = 1, 2, \dots, N; M \leftarrow m; \mathbf{break};$
13. **end**
14. **end**

maximum number of iterations. Note that in m^{th} iteration, the fine evaluation in line 5 is performed from m to N , not 1 to N . This is because the coarse values in the previous $(m-1)$ steps have been corrected by the fine solver in the previous $(m-1)$ iterations.

The proposed DT-based Parareal method has the following advantages: 1) the DT method is more efficient than numerical methods since it can reduce the sequential time and improve the overall speedup; 2) the DT method, at the same time, is more accurate than numerical methods since the coarse solution by the DT method is closer to the true solution and fewer iterations are needed to make the Parareal method converge; and 3) the DT method is highly flexible since the order and the time step can be adjusted adaptively (as shown in the next section) to balance the accuracy and efficiency of the coarse solver during the simulation, thus, improving the overall performance.

Algorithm 3 VOVS-DT Strategy

input: $t_0, t_{end}, \mathbf{x}_0, h_0, K_0$
output: $\mathbf{x}(t_n), \mathbf{h}(t_n), \mathbf{K}(t_n), t_0 \leq t_n \leq t_{end}$
offline stage: derive the recursive equation $\mathbf{X}(k+1) = \mathbf{F}(\mathbf{X}(0:k))$
online stage:

1. $\mathbf{x}(t_0) \leftarrow \mathbf{x}_0, \mathbf{h}(t_0) \leftarrow h_0, \mathbf{K}(t_0) \leftarrow K_0$
2. $t \leftarrow t_0, h \leftarrow h_0, K \leftarrow K_0, \mathbf{X}(0) \leftarrow \mathbf{x}_0, n \leftarrow 0$
3. **while** $t+h < t_{end}$
4. **for** $k = 0 : K$ // computer power series coefficients
5. $\mathbf{X}(k+1) = \mathbf{F}(\mathbf{X}(0:k));$
6. **end**
7. $\mathbf{x}(t_{n+1}) \leftarrow \sum_{k \in \{0,1,\dots,K\}} \mathbf{X}(k) h^k$
8. $err \leftarrow \|\mathbf{X}(K+1) h^{K+1}\|_{\infty}$ // estimation of error
9. **if** $err < \varepsilon_1; h \leftarrow \min(q_1 h, h_{\max}); \mathbf{end}$
10. **if** $err > \varepsilon_2; h \leftarrow \max(q_2 h, h_{\min}); \mathbf{end}$
11. **if** $err < \varepsilon_3; K \leftarrow \max(K - \Delta K, K_{\min}); \mathbf{end}$
12. **if** $err > \varepsilon_4; K \leftarrow \min(K + \Delta K, K_{\max}); \mathbf{end}$
13. $\mathbf{h}(t_{n+1}) \leftarrow h, \mathbf{K}(t_{n+1}) \leftarrow K, t \leftarrow t+h, n \leftarrow n+1$
14. **end**

C. ADAPTIVE STRATEGIES FOR DT-BASED PARAREAL METHOD

1) VOVS-DT STRATEGY

Algorithm 3 insert shows the proposed VOVS-DT strategy, where the outputs include not only the trajectories of state variables but also the variable time step lengths $\mathbf{h}(t_n)$ and variable orders $\mathbf{K}(t_n)$ during the simulation; lines 1 to 7 are similar to the DT method in Algorithm 1, except for the initialization of additional variables associated with the time step length and the order of DT; line 8 is error estimation using the $(K+1)^{\text{th}}$ power series term; lines 9 to 12 are the adaptive change of time step length and the order of DT based on the error estimation. In Algorithm 3, $\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4$ are pre-defined error thresholds; q_1, q_2 are the factors to adjust the time step length; ΔK is an incremental adjustment of the order of DT; h_{\max}, h_{\min} are the pre-defined maximum and minimum time step length; K_{\max}, K_{\min} are the maximum and minimum order.

The proposed VOVS-DT strategy has the following compelling advantages when used as the coarse solver in the Parareal method: 1) it increases the order and reduces the time step length of the DT-based coarse solver when the error is larger than a threshold during the simulation, which could reduce the needed number of iterations for the Parareal method to converge; 2) it decreases the order and increases the time step length of the DT-based coarse solver when the error is smaller than a threshold during the simulation, which might slightly increase the number of iterations but can avoid unnecessary computation burden and improve the overall efficiency. Note that these advantages are not easily attainable from other existing methods.

Algorithm 4 VW-Parareal Strategy

input: $t_0, t_{end}, h_w, \mathbf{x}(t_0)$
output: $\mathbf{x}(t), t_0 \leq t \leq t_{end}; \mathbf{H}; \mathbf{M}$

1. $t \leftarrow t_0, \mathbf{H} \leftarrow h_w, \mathbf{M} \leftarrow \emptyset, \mathbf{x}(t) \leftarrow \mathbf{x}(t_0)$
 2. **while** $t + h_w < t_{end}$
 3. $t_1^w \leftarrow t, t_2^w \leftarrow t + h_w$
 4. call **Algorithm 2:** DT-based Parareal method to compute $\mathbf{x}_1, \dots, \mathbf{x}_N; M$ for the time interval $[t_1^w, t_2^w]$
 5. **if** $M < M_1$, **then** $h_w \leftarrow \min(q_3 h_w, h_{w,max})$
 6. **if** $M > M_2$, **then** $h_w \leftarrow \max(q_4 h_w, h_{w,min})$
 7. $t \leftarrow t + h_w, \mathbf{H} \leftarrow [\mathbf{H}, h_w], \mathbf{M} \leftarrow [\mathbf{M}, M], \mathbf{x}(t) \leftarrow [\mathbf{x}(t), \mathbf{x}_1, \dots, \mathbf{x}_N]$
 8. **end**
-

2) VW-PARAREAL STRATEGY

To overcome the divergence issue of the Parareal method for long simulation period lengths, a variable window strategy is proposed that can change the window size adaptively according to the needed number of iterations in the previous window. Algorithm 4 insert shows the detailed procedure of the proposed VW-Parareal strategy, where the outputs include not only the trajectory of state variables but also the variable window lengths \mathbf{H} and the number of iterations \mathbf{M} in each window during the simulation; line 1 is the initialization of variables; line 3 defines a simulation window; line 4 performs Parareal simulation over this window; lines 5 and 6 adjust the window length of the next window according to the number of iterations in the current window; line 7 saves the results. In Algorithm 4, M_1, M_2 are the pre-defined thresholds of the number of iterations; q_3, q_4 are the factors to adjust the window length; $h_{w,max}, h_{w,min}$ are the pre-defined maximum and minimum window lengths, respectively.

The proposed VW-Parareal strategy has the following advantages. First, the use of the multi-window strategy makes it easier for the Parareal method to converge because the length of a window is smaller than the total simulation length. Second, the proposed VW strategy can provide more suitable window sizes during the simulation than a pre-defined fixed size, because the Parareal method with a fixed window size may diverge when the size is too large and may be inefficient when the size is too small. Therefore, the proposed VW-Parareal strategy can balance the convergence with efficiency performance.

3) IMPROVED COARSE SOLUTION UPDATE STRATEGY

The coarse solution update strategy in line 9 of Algorithm 2, i.e., $\mathbf{x}_n^{*,m} \leftarrow \mathbf{x}_n^{f,m} + (\mathbf{x}_n^{c,m} - \mathbf{x}_n^{c,m-1})$, could be interpreted as a first-order Newton iteration method to solve the multi-shooting problem, where the Jacobian matrix in the Newton iteration method is approximated by the coarse solutions [30]. Following the interpretation in [30], this section extends the Parareal method to the second-order Newton iteration method to solve the multi-shooting problem,

where both the Jacobian matrix in the first-order term and the Hessian matrix in the second-order term are approximated by the coarse solutions. The procedure is similar to [30] and the details are omitted. The coarse solution update strategy including the second-order term can be represented as follow:

$$\mathbf{x}_n^{*,m} \leftarrow \mathbf{x}_n^{f,m} + \underbrace{(\mathbf{x}_n^{c,m} - \mathbf{x}_n^{c,m-1})}_{\text{first-order term}} + \underbrace{0.5(\mathbf{x}_n^{c,m+1} - 2\mathbf{x}_n^{c,m} + \mathbf{x}_n^{c,m-1})}_{\text{second-order term}} \quad (3)$$

Here, note that the value of $\mathbf{x}_n^{c,m+1}$ is unknown in the m^{th} iteration. In this work, it is approximated by $\mathbf{x}_n^{c,m+1} \approx \mathbf{x}_n^{c,m}$, which results in the proposed improved coarse update strategy.

$$\mathbf{x}_n^{*,m} \leftarrow \mathbf{x}_n^{f,m} + 0.5(\mathbf{x}_n^{c,m} - \mathbf{x}_n^{c,m-1})$$

Remark: The proposed improved coarse solution update strategy assumes $\mathbf{x}_n^{c,m+1} \approx \mathbf{x}_n^{c,m}$. This assumption is motivated by the intuitive observation that the difference between $\mathbf{x}_n^{c,m+1}$ and $\mathbf{x}_n^{c,m}$ tends to decrease with the increase of the number of iterations if the Parareal method converges. This simple yet practically effective modification improves the coarse solution update strategy and, thus, can reduce the number of Parareal iterations if the Parareal method converges. Despite its practical effectiveness, rigorous validation of this assumption and the proposed improved coarse solution update strategy is also important and necessary to advance the understanding of the temporal parallelization algorithms and would be an interesting research topic that deserves further exploration.

IV. CASE STUDY

The proposed DT-based adaptive Parareal method is first tested on the IEEE 39-bus system [26] with classical generator models, and the Polish 2383-bus system [31] with detailed models of generators, exciters, governors, and turbines. Then, various scenarios are tested to demonstrate the reliable performance of the proposed method. In each test, the following three methods are compared: 1) M1: the traditional Parareal method, 2) M2: the proposed DT-based Parareal method, and 3) M3: the proposed DT-based adaptive Parareal method. Compared with M2, M3 further integrates the proposed adaptive strategies. In all three methods, the fine solvers are selected as the 4th order Runge-Kutta method.

For all the tests in Section IV-A and Section IV-B, the test settings are the following. The time step lengths of the coarse and fine solver are selected as 0.1 seconds and 0.001 second, respectively; the parameters in the algorithms are selected as $h_{max} = 0.2, h_{min} = 0.0001, K_{max} = 20, K_{min} = 2, q_1 = 1.1, q_2 = 0.9, \Delta K = 1, m_{max} = 15, tol = 0.1$. For the 39-bus system, the simulated contingency is a fictitious temporary three-phase fault at bus 8 applied at $t = 1$ second that clears after 0.2 seconds. For the 2383-bus system, the simulated contingency is a permanent three-phase fault at bus 9 applied

at $t = 1$ second and cleared after 0.4 seconds by tripping the line between bus 6 and bus 9.

Note that these two specific scenarios are used to provide detailed simulation results and corresponding descriptions. A wide range of different scenarios to further validate the proposed method is considered in Section IV-C. The tests are conducted in MATLAB R2021b environment on a personal computer with i5-8250U CPU. Both the DT method and the Parareal method are implemented in MATLAB using the research code developed by the authors. The Matpower toolbox is used to solve the power flow and initialize the dynamic simulation.

A. PERFORMANCE WITH THE 39-BUS SYSTEM

Table 1 gives the number of iterations and CPU time for the three methods when the Parareal window size is 0.75 seconds. It shows the traditional Parareal method takes an average of 7 iterations in each window to converge while the proposed DT-based Parareal method and the DT-based adaptive Parareal method need only one iteration in each window. Moreover, the computation time of the DT-based Parareal method and the DT-based adaptive Parareal method is reduced by 37.5% and 50% respectively, compared to the traditional Parareal method.

TABLE 1. Number of iterations and CPU time in the 39-bus system test.

Method	No. Iterations	CPU Time (seconds)
M1	7	16.12
M2	1	10.33
M3	1	8.01

When the Parareal window size is further increased from 0.75 seconds to 1 second, both the traditional Parareal method and the DT-based Parareal method diverge, while the DT-based adaptive Parareal method can still converge. For the DT-based adaptive Parareal method, Fig. 4 shows the trajectories of rotor angles of all generators, voltages of all buses, and electrical power outputs of all generators. Fig. 5 shows the window length and the number of iterations of each Parareal window. Fig. 6 shows the time step length and order of the DT-based coarse solver. It can be seen from these results that: 1) the first Parareal window needs 11 iterations to converge, and 2) the proposed DT-based adaptive Parareal method can adaptively adjust the window size, order, and time step length of the coarse solver during the simulation such that convergence is achieved, as the number of iterations is reduced in the subsequent windows (as shown also with the CPU time in Table 1).

B. PERFORMANCE WITH THE 2383-BUS SYSTEM

Table 2 gives the number of iterations and CPU time of the three methods when the Parareal window size is 0.75 seconds. The traditional Parareal method diverges while both the proposed DT-based Parareal method and the DT-based

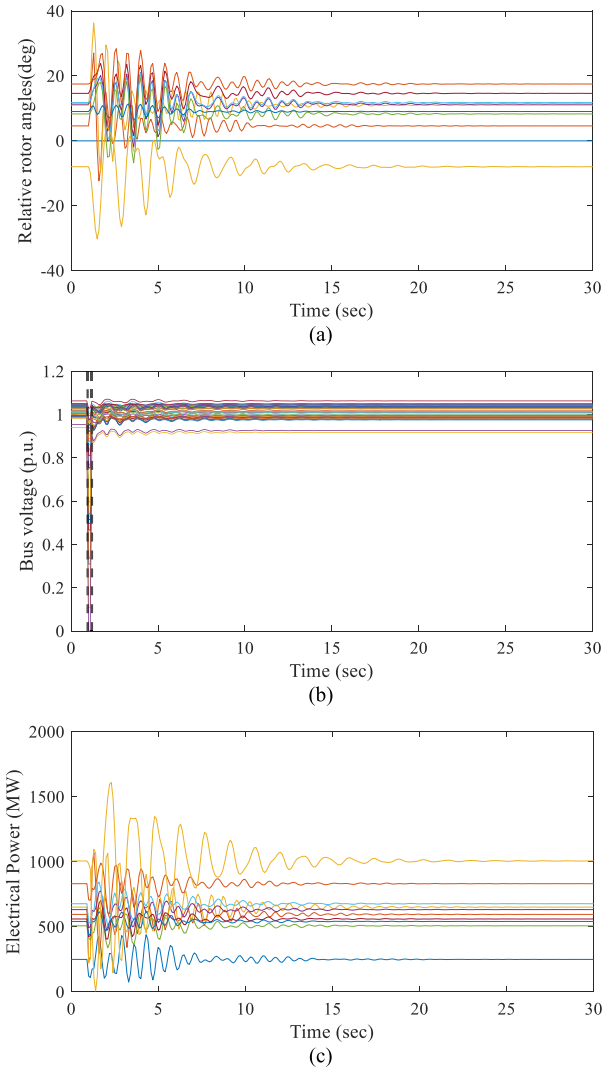


FIGURE 4. Trajectories of state variables in the 39-bus system test using the DT-based adaptive Parareal method: a) rotor angles of all generators; b) bus voltages of all buses, and c) electrical power outputs of all generators.

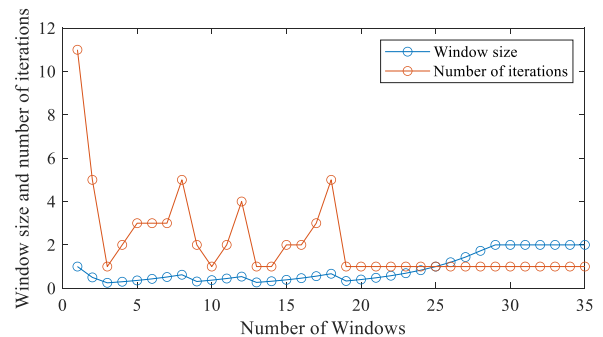


FIGURE 5. The window size and the number of iterations of each window in the 39-bus system test using the DT-based adaptive Parareal method.

adaptive Parareal method converge and need only one iteration in each window. Since the DT-based adaptive Parareal method adjusts the window size, the order, and the time step

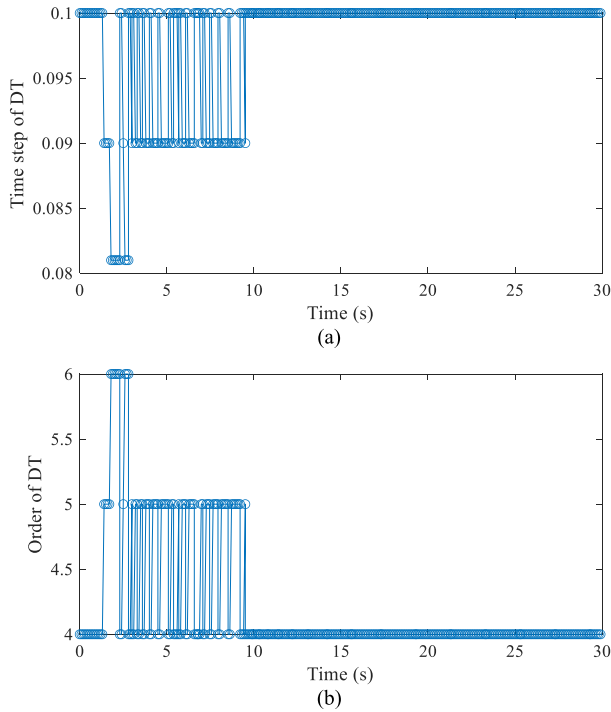


FIGURE 6. The time step length and order of the DT-based coarse solver in the 39-bus system test using the DT-based adaptive Parareal method: a) the time step length, b) the order.

TABLE 2. The number of iterations and CPU time in the 2383-bus system test.

Method	No. Iterations	CPU Time (seconds)
M1	Divergent	N/A
M2	1	158.46
M3	1	117.60

length adaptively during the simulation, the computation time is reduced by 25% compared with the DT-based Parareal method without adaptive strategies.

When the Parareal window size is further increased from 0.75 seconds to 1 second, the results are similar to the 39-bus system test, i.e., only the DT-based adaptive Parareal method converges. Figs. 7 to 9 respectively give the trajectories of rotor angles of all generators, voltages of all buses, and electrical power outputs of all generators; the window length and the number of iterations of each Parareal window; and the time step length and order of the DT-based coarse solver. From Figs. 8 and 9, it can be seen that the number of iterations is 12 in the first window but it is reduced in the subsequent windows benefiting from the flexibility of adaptively adjusting the window size, the order, and the time step length of the coarse solver during the simulation. These results demonstrate the better convergence performance of the proposed DT-based adaptive Parareal method in a large power system with detailed dynamic models.

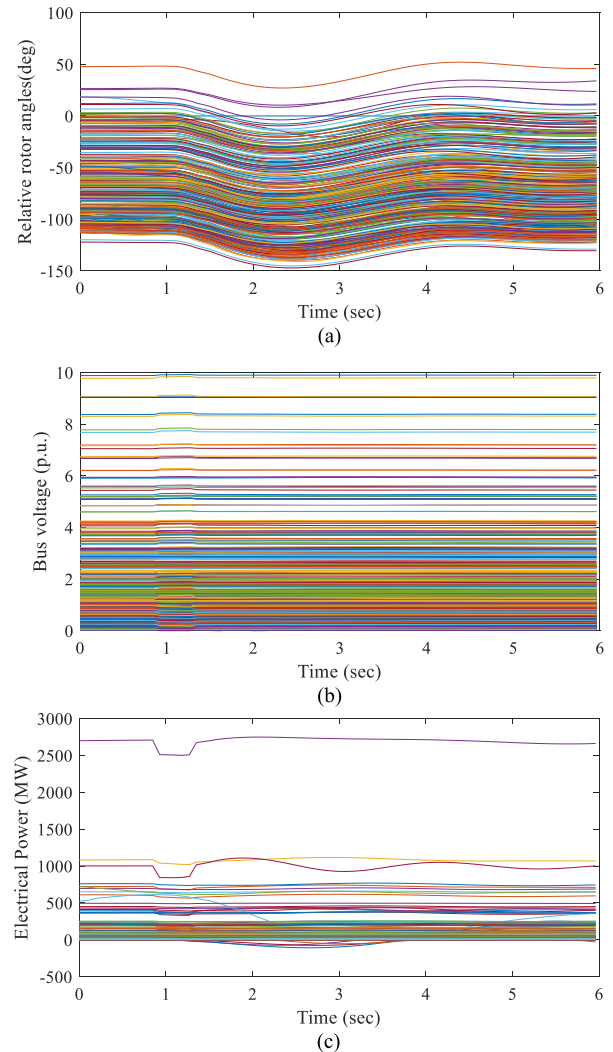


FIGURE 7. Trajectories of state variables in the 2383-bus system test using the DT-based adaptive Parareal method: a) rotor angles of all generators; b) bus voltages of all buses, and c) electrical power outputs of all generators.

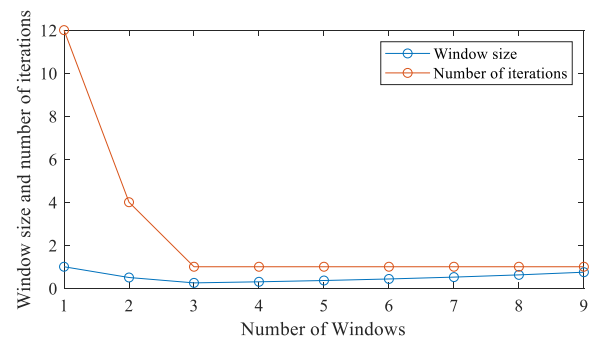


FIGURE 8. The window size and the number of iterations of each window in the 2383-bus system test using the DT-based adaptive Parareal method.

C. RELIABLE PERFORMANCE IN VARIOUS SCENARIOS

In this study, the three methods M1, M2, and M3 are tested on the 39-bus system under 54 scenarios in total, with

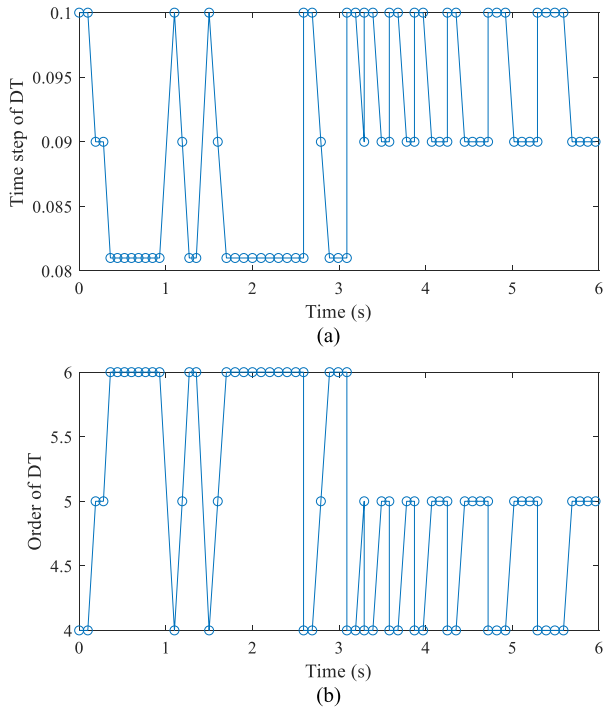


FIGURE 9. The time step length and order of the DT-based coarse solver in the 2383-bus system test using the DT-based adaptive Parareal method: a) the time step length, b) the order.

different contingencies and parameter settings, as follows: 1) Three contingencies with increased severity, i.e., Contingency #1: a temporary fault at bus 8, cleared after 0.1 seconds; Contingency #2: a permanent fault at bus 8, cleared after 0.2 seconds by tripping the branch between bus 8 and bus 9, and Contingency #3: a permanent fault at bus 8, cleared after 0.3 seconds by tripping the branch between bus 8 and bus 9; and 2) Different time step length of the coarse solver, i.e., $h_c = 0.1$ seconds, 0.08 seconds, and 0.05 seconds. 3) Different Parareal window size, i.e., $h_w = 0.5$ seconds, 0.75 seconds, 1.0 second, 1.25 seconds, 1.5 seconds, and 2 seconds.

Table 3 to Table 5 give the average number of iterations in each window, respectively for the three methods. They show that there are 23 divergent scenarios (42.5% of the total number of scenarios) for the traditional Parareal method and 4 divergent scenarios (7.4% of the total number of scenarios) for the proposed DT-based Parareal method; while the proposed DT-based adaptive Parareal method converges in all the 54 scenarios. These results demonstrate the effectiveness of the proposed DT-based adaptive Parareal method in improving the convergence performance and robustness across various scenarios.

A similar study is conducted on the Polish 2383-bus system. First, the performance under severe contingencies is validated by creating three contingencies with increasing severities, where contingency #3 leads to unstable trajectories: 1) Contingency #1 has a temporary fault at bus 9 cleared

TABLE 3. Average iterations of a traditional parareal method.

--	Contingency # 1	Contingency # 2	Contingency # 3	
$h_c = 0.100$	$h_w = 0.50$	1	1	5
	$h_w = 0.75$	7	7	7
	$h_w = 1.00$	Divergent	Divergent	Divergent
	$h_w = 1.25$	Divergent	Divergent	Divergent
	$h_w = 1.50$	Divergent	Divergent	Divergent
	$h_w = 2.00$	Divergent	Divergent	Divergent
$h_c = 0.080$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	Divergent
	$h_w = 1.25$	Divergent	Divergent	Divergent
	$h_w = 1.50$	Divergent	Divergent	Divergent
	$h_w = 2.00$	Divergent	Divergent	Divergent
$h_c = 0.050$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	1
	$h_w = 1.25$	1	1	1
	$h_w = 1.50$	1	1	1
	$h_w = 2.00$	1	1	Divergent

TABLE 4. Average iterations of the DT-based parareal method.

--	Contingency # 1	Contingency # 2	Contingency # 3	
$h_c = 0.100$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	1
	$h_w = 1.25$	1	1	1
	$h_w = 1.50$	1	1	Divergent
	$h_w = 2.00$	Divergent	Divergent	Divergent
$h_c = 0.08$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	1
	$h_w = 1.25$	1	1	1
	$h_w = 1.50$	1	1	1
	$h_w = 2.00$	1	1	1
$h_c = 0.050$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	1
	$h_w = 1.25$	1	1	1
	$h_w = 1.50$	1	1	1
	$h_w = 2.00$	1	1	1

after 0.4 seconds; Contingency #2 has a permanent fault at bus 9 cleared after 0.4 seconds by tripping the branch between bus 6 and bus 9, and Contingency #3 has a permanent fault

TABLE 5. Average iterations of the DT-based adaptive parareal method.

--		Contingency # 1	Contingency # 2	Contingency # 3
$h_c = 0.100$	$h_w = 0.50$	1	1.31	1.81
	$h_w = 0.75$	1	1.80	2.00
	$h_w = 1.00$	1	2.22	2.08
	$h_w = 1.25$	1	2.13	2.17
	$h_w = 1.50$	1	2.29	2.57
	$h_w = 2.00$	1	2.50	2.50
$h_c = 0.08$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	1
	$h_w = 1.25$	1	1	1
	$h_w = 1.50$	1	1	1
	$h_w = 2.00$	1	1	1
$h_c = 0.050$	$h_w = 0.50$	1	1	1
	$h_w = 0.75$	1	1	1
	$h_w = 1.00$	1	1	1
	$h_w = 1.25$	1	1	1
	$h_w = 1.50$	1	1	1
	$h_w = 2.00$	1	1	1

at bus 9 cleared after 0.6 seconds by tripping the branch between bus 6 and bus 9. None of the three cases is converged with the conventional Parareal method. In contrast, the proposed method converges and takes an average of 2.56, 2.56, and 3.46 iterations, respectively, for the three contingencies. Second, the impact of parameter settings is validated by testing different time step lengths of the coarse solver, i.e., $h_c = 0.1$ seconds and 0.08 seconds, respectively, and different Parareal window sizes, i.e., $h_w = 1.0$ seconds and 1.25 seconds, respectively. Similar results are obtained where the conventional Parareal method diverges with those large time step lengths for the coarse solver and window lengths while the proposed method converges by less than 5 iterations (averagely 1.42 and 2.56 iterations when $h_c = 0.1$ seconds and 0.08 seconds, respectively; and 2.56 and 4.57 iterations when $h_w = 1.0$ second and 1.25 seconds, respectively). These results show the proposed DT-based Adaptive Parareal method has reliable performance on the Polish 2383-bus system as well.

In the above tests, the tolerance $tol=0.1$ is selected in the Parareal algorithm. However, the proposed method could still converge for a smaller tolerance while the conventional method would diverge. Two cases are tested on the Polish 2383-bus system with $tol=0.1$ and $tol=0.01$, respectively. The conventional Parareal method diverges in both cases while the proposed method converges in an average of 2.56 and 3.25 iterations, respectively.

V. CONCLUSION

The Parareal-based parallel-in-time method and the differential transformation-based semi-analytical solution method are two of the state-of-the-art techniques in power system time domain simulation. Leveraging the temporal parallelization capability of the Parareal method and the high accuracy, efficiency, and adaptivity of the differential transformation method, this paper proposes a differential transformation-based variable-order variable-step variable-window adaptive Parareal method for temporal parallelization of power system simulation with greatly enhanced convergence performance and efficiency. Since the coarse solver plays a key role in the performance of the Parareal method, this paper employs the DT method as the coarse solver in the Parareal method and further designs a set of adaptive strategies to enhance the performance. Extensive simulations on a 39-bus system and a 2383-bus system demonstrate that the proposed approach improves the convergence performance and computational efficiency of the Parareal method and has reliable performance under various contingencies and parameter settings.

APPENDIX

The motivation, definition, and selected transformation rules of the DT method are summarized below, and more details could be found in [23], [24].

Motivation: The DT method is a promising semi-analytical solution method that was originally proposed in the applied mathematics field and introduced to the power system field recently by [23]. It aims at deriving an approximate solution of nonlinear differential equations in the form of finite power series of time with any desired order. Different from other SAS methods such as the Adomian decomposition method [20], [21] and the power series method [22], a unique benefit of the DT method is that it provides various transformation rules to convert a nonlinear function directly to its power series coefficients. As a result, the DT method avoids the complicated process of calculating explicit high-order derivatives and provides a flexible and efficient procedure for deriving SAS up to any desired order.

Definition: The DT of a real continuous function $x(t)$, $t \in \mathbb{R}$ is defined by (A1), where $k \in \mathbb{N}$ is the order.

$$X(k) = \frac{1}{k!} \left[\frac{d^k x(t)}{dt^k} \right]_{t=0} \quad (\text{A1})$$

Transformation rules: The DT method provides transformation rules for various generic functions (both linear and nonlinear functions; both simple and compositional functions). The widely used transformation rules in the power system model are summarized below. Note that these rules apply to not only scalar functions but also vector-valued functions.

$$X(0) = x(0) \quad (\text{A2})$$

$$y(t) = cx(t) \rightarrow Y(k) = cX(k) \quad (\text{A3})$$

$$z(t) = x(t) \pm y(t) \rightarrow Z(k) = X(k) \pm Y(k) \quad (\text{A4})$$

$$z(t) = x(t)y(t) \rightarrow Z(k) = \sum_{m=0}^k X(m)Y(k-m) \quad (\text{A5})$$

$$y(t) = t^n \rightarrow Y(k) = \varrho(k-n) = \begin{cases} 1, & k = n \\ 0, & k \neq n \end{cases} \quad (\text{A6})$$

$$y(t) = c \rightarrow Y(k) = c\varrho(k) = \begin{cases} c, & k = 0 \\ 0, & k \neq 0 \end{cases} \quad (\text{A7})$$

$$y(t) = \frac{dx(t)}{dt} \rightarrow Y(k) = (k+1)X(k+1) \quad (\text{A8})$$

$$\begin{cases} \phi(t) = \sin \delta(t) \\ \psi(t) = \cos \delta(t) \end{cases} \rightarrow \begin{cases} \Phi(k) = \sum_{m=0}^{k-1} \frac{k-m}{k} \Psi(m) \Delta(k-m) \\ \Psi(k) = -\sum_{m=0}^{k-1} \frac{k-m}{k} \Phi(m) \Delta(k-m) \end{cases} \quad (\text{A9})$$

$$y(t) = e^{x(t)} \rightarrow Y(k) = \frac{1}{k} \sum_{m=0}^{k-1} (k-m)Y(m)X(k-m) \quad (\text{A10})$$

$$y(t) = \sqrt{x} \rightarrow Y(k) = \frac{1}{2Y(0)}X(k) - \frac{1}{2Y(0)} \sum_{m=1}^{k-1} Y(m)Y(k-m) \quad (\text{A11})$$

$$z(t) = x(t)/y(t) \rightarrow Z(k) = \frac{1}{Y(0)}X(k) - \frac{1}{Y(0)} \sum_{m=0}^{k-1} Z(m)Y(k-m) \quad (\text{A12})$$

ACKNOWLEDGMENT

The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a non-exclusive, paid-up, irrevocable, and world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for the U.S. Government purposes. The Department of Energy (DOE) will provide public access to these results of federally sponsored research in accordance with the DOE public access plan.

REFERENCES

- [1] Z. Qin, X. Chen, Y. Hou, H. Liu, and Y. Yang, "Coordination of preventive, emergency and restorative dispatch in extreme weather events," *IEEE Trans. Power Syst.*, vol. 37, no. 4, pp. 2624–2638, Jul. 2022.
- [2] Y. Liu, K. Sun, B. Wang, R. Yao, and W. Kang, "A sparse grid scheme for fast transient stability simulation with reduced redundancy," in *Proc. IEEE Milan PowerTech*, Jun. 2019, pp. 1–6.
- [3] P. Kundur, N. J. Balu, and M. G. Lauby, *Power System Stability and Control*. New York, NY, USA: McGraw-Hill, 1994.
- [4] F. Milano, *Power System Modelling and Scripting*. Cham, Switzerland: Springer, 2010.
- [5] J. H. Chow, *Power System Coherency and Model Reduction*. Cham, Switzerland: Springer, 2013.
- [6] D. Osipov and K. Sun, "Adaptive nonlinear model reduction for fast power system simulation," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6746–6754, Nov. 2018.
- [7] S. Zadkhast, J. Jatskevich, and E. Vaahedi, "A multi-decomposition approach for accelerated time-domain simulation of transient stability problems," *IEEE Trans. Power Syst.*, vol. 30, no. 5, pp. 2301–2311, Sep. 2015.
- [8] P. Aristidou, S. Lebeau, and T. Van Cutsem, "Power system dynamic simulations using a parallel two-level schur-complement decomposition," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3984–3995, Sep. 2016.
- [9] P. Aristidou, D. Fabozzi, and T. Van Cutsem, "Dynamic simulation of large-scale power systems using a parallel Schur-complement-based decomposition method," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 10, pp. 2561–2570, Oct. 2014.
- [10] M. A. Tomim, J. R. Marti, and J. A. Passos Filho, "Parallel transient stability simulation based on multi-area Thévenin equivalents," *IEEE Trans. Smart Grid*, vol. 8, no. 3, pp. 1366–1377, May 2017.
- [11] Y. Liu and Q. Jiang, "Two-stage parallel waveform relaxation method for large-scale power system transient stability simulation," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 153–162, Jan. 2016.
- [12] G. Gurrara, A. Dimitrovski, S. Pannala, S. Simunovic, and M. Starke, "Parareal in time for fast power system dynamic simulations," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 1820–1830, May 2016.
- [13] N. Duan, A. Dimitrovski, S. Simunovic, and K. Sun, "Applying reduced generator models in the coarse solver of parareal in time parallel power system simulation," in *Proc. IEEE PES Innov. Smart Grid Technol. Conf. Eur. (ISGT-Eur.)*, Oct. 2016, pp. 1–5.
- [14] N. Duan, A. Dimitrovski, S. Simunovic, K. Sun, J. Qi, and J. Wang, "Embedding spatial decomposition in parareal in time power system simulation," in *Proc. IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf. (ISGT)*, Feb. 2018, pp. 1–6.
- [15] N. Duan, S. Simunovic, A. Dimitrovski, and K. Sun, "Improving the convergence rate of Parareal-in-time power system simulation using the Krylov subspace," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, Aug. 2018, pp. 1–5.
- [16] D. Osipov, N. Duan, A. Dimitrovski, S. Allu, S. Simunovic, and K. Sun, "Adaptive model reduction for parareal in time method for transient stability simulations," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, Aug. 2018, pp. 1–5.
- [17] D. Osipov, N. Duan, S. Allu, S. Simunovic, A. Dimitrovski, and K. Sun, "Distributed parareal in time with adaptive coarse solver for large scale power system simulations," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, Aug. 2019, pp. 1–5.
- [18] B. Park et al., "Performance and feature improvements in parareal-based power system dynamic simulation," in *Proc. IEEE Int. Conf. Power Syst. Technol. (POWERCON)*, Sep. 2020, pp. 1–6.
- [19] B. Park, K. Sun, A. Dimitrovski, Y. Liu, and S. Simunovic, "Examination of semi-analytical solution methods in the coarse operator of parareal algorithm for power system simulation," *IEEE Trans. Power Syst.*, vol. 36, no. 6, pp. 5068–5080, Nov. 2021.
- [20] N. Duan and K. Sun, "Power system simulation using the multistage adomian decomposition method," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 430–441, Jan. 2017.
- [21] G. Gurrara, D. L. Dinesha, A. Dimitrovski, P. Sreekanth, S. Simunovic, and M. Starke, "Large multi-machine power system simulations using multi-stage Adomian decomposition," *IEEE Trans. Power Syst.*, vol. 32, no. 5, pp. 3594–3606, Sep. 2017.
- [22] B. Wang, N. Duan, and K. Sun, "A time–power series-based semi-analytical approach for power system simulation," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 841–851, Mar. 2019.
- [23] Y. Liu, K. Sun, R. Yao, and B. Wang, "Power system time domain simulation using a differential transformation method," *IEEE Trans. Power Syst.*, vol. 34, no. 5, pp. 3739–3748, Sep. 2019.
- [24] Y. Liu and K. Sun, "Solving power system differential algebraic equations using differential transformation," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 2289–2299, May 2020.
- [25] R. Yao, Y. Liu, K. Sun, F. Qiu, and J. Wang, "Efficient and robust dynamic simulation of power systems with holomorphic embedding," *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 938–949, Mar. 2020.
- [26] R. Diaio et al., "On parallelizing single dynamic simulation using HPC techniques and APIs of commercial software," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 2225–2233, May 2017.
- [27] A. S. Nielsen, "Feasibility study of the parareal algorithm," M.S. thesis, Dept. Inform. Math. Model., Univ. Denmark, Lyngby, Denmark, 2012.
- [28] L. Baffico et al., "Parallel in time molecular dynamics simulations," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 66, no. 5, pp. 1–4, Nov. 2002.

- [29] Y. Maday and O. Mula, "An adaptive parareal algorithm," *J. Comput. Appl. Math.*, vol. 377, Oct. 2020, Art. no. 112915.
- [30] M. J. Gander and S. Vandewalle, "Analysis of the parareal time-parallel time-integration method," *SIAM J. Sci. Comput.*, vol. 29, no. 2, pp. 556–578, 2007.
- [31] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MAT-POWER: Steady-state operations, planning and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.



YANG LIU (Member, IEEE) received the B.S. degree in energy and power engineering from Xi'an Jiaotong University, China, in 2013, the M.S. degree in power engineering from Tsinghua University, China, in 2016, and the Ph.D. degree in electrical engineering from The University of Tennessee Knoxville, Knoxville TN, USA, in 2021. He was a Postdoctoral Research Associate at the University of Tennessee, Knoxville, TN, USA, in 2022. Currently, he is a Postdoctoral Appointee with Argonne National Laboratory. His research interests include power system simulation, dynamics, stability, and control.



BYUNGKWON PARK (Member, IEEE) received the B.S. degree in electrical engineering from Chonbuk National University, South Korea, in 2011, and the M.S. and Ph.D. degrees in electrical engineering from the University of Wisconsin–Madison, Madison, WI, USA, in 2014 and 2018, respectively. He is currently an Assistant Professor with the Department of Electrical Engineering, Soongsil University (SSU), Seoul, South Korea. Before joining SSU, he was a Research Staff with the Computational Sciences and Engineering Division, Oak Ridge National Laboratory. His research interests include modeling, simulation, control, and optimization of electrical energy systems.



KAI SUN (Senior Member, IEEE) received the B.S. degree in automation and the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 1999 and 2004, respectively. He was a Project Manager in grid operations and planning with the Electric Power Research Institute (EPRI), Palo Alto, CA, USA, from 2007 to 2012. He is currently a Professor at the Department of EECS, The University of Tennessee Knoxville, Knoxville TN, USA. He is an Associate Editor with IEEE TRANSACTIONS ON POWER SYSTEMS and IEEE OPEN ACCESS JOURNAL OF POWER AND ENERGY.



ALEKSANDAR DIMITROVSKI (Senior Member, IEEE) received the B.Sc. degree in electrical engineering with emphasis in power, the M.Sc. degree in applied computer sciences, and the Ph.D. degree in electrical engineering with emphasis in power from Europe. He is currently an Associate Professor with the Electrical and Computer Engineering Department, University of Central Florida. Before joining UCF, he had worked in research institutions, power industry, and academia, both in the USA and Europe. His research interests include modeling, analysis, protection, and control of uncertain power systems.



SRDJAN SIMUNOVIC received the M.Sc. and Ph.D. degrees in civil engineering from Carnegie Mellon University, Pittsburgh, PA, USA. He is currently a Distinguished Research Staff with the Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA. His research interests include applied mathematics for multiscale and multiphysics problems, parallel algorithms and computing, energy storage, and computational modeling of materials and processes.

...