5-2018

Towards Faster-than-real-time Power System Simulation Using a Semi-analytical Approach and High-performance Computing

Nan Duan
University of Tennessee, nduan@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Nan Duan entitled "Towards Faster-than-real-time Power System Simulation Using a Semi-analytical Approach and High-performance Computing." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Kai Sun, Major Professor

We have read this dissertation and recommend its acceptance:

Fangxing Li, Kevin L. Tomsovic, Xiaopeng Zhao

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Towards Faster-than-real-time Power System Simulation Using a Semi-analytical Approach and High-performance Computing

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Nan Duan

May 2018

DEDICATION

I dedicate this dissertation to my mother Hong Yang and my father Changming Duan and my wife Ran Huang.

ACKNOWLEDGEMENTS

In the past four and half years of my Ph.D. program I have received the help from numerous people. I cannot finish the very challenging doctoral research without their guidance and help. I feel very lucky to have the chance to learn from them, not only how to conduct research but how to be a better person, how to face the day-to-day challenge of life.

I would like to thank my advisor Dr. Kai Sun for guiding me through the doctoral study. Since I was not from power system background, I had to familiarize myself with power system knowledges. Dr. Kai Sun is always very encouraging when I face difficulties. He also teaches me how to balance research and life. He wants me to be not only well trained for a successful career but also mindful about a healthy and sustainable life style. I appreciate his help very much.

I also would like to thank all my colleagues that I have worked with at UTK, Dr. Fengkai Hu, Dr. Bin Wang, Dr. Weihong Huang, Wenyun Ju, Denis Osipov, Hira Amna Saleem, Yongli Zhu, Xin Xu, Yang Liu, Tianwei Xia, Dr. Rui Yao, Dr. Chengxi Liu, Dr. Junjian Qi, Dr. Miao Fan. I appreciate the inspiring discussions I had with them. I learn exciting new ideas from my colleagues every day, the interactions I had with them will serve as a source of creativity for my future research. I'm very fortunate to have so many talented colleagues.

Finally, I would like to express my appreciation to my dissertation committee, Dr. Kai Sun, Dr. Kevin Tomsovic, Dr. Fangxing (Fran) Li and Dr. Xiaopeng Zhao for providing valuable suggestions for improving the quality of my dissertation.

ABSTRACT

This dissertation investigates two possible directions of achieving faster-than-real-time simulation of power systems. The first direction is to develop a semi-analytical solution which represents the nonlinear dynamic characteristics of power systems in a limited time period. The second direction is to develop a parallel simulation scheme which allows the local numerical solutions of power systems to be developed independently in consecutive time intervals and then iteratively corrected toward the accurate global solution through the entire simulation time period.

For the first direction, the semi-analytical solution is acquired using Adomian decomposition method (ADM). The ADM assumes the analytical solution of any nonlinear system can be decomposed into the summation of infinite analytical expressions. Those expressions are derived recursively using the system differential equations. By only keeping a finite number of those analytical expressions, an approximation of the analytical solution is yielded, which is defined as a semi-analytical solution. The semi-analytical solutions can be developed offline and evaluated online to facilitate the speedup of simulations. A parallel implementation and variable time window approach for the online evaluation stage are proposed in addition to the time performance analysis.

For the second direction, the Parareal-in-time algorithm is tested for power system simulation. Parareal is essentially a multiple shooting method. It decomposes the simulation time into coarse time intervals and then fine time intervals within each coarse interval. The numerical integration uses a computational cheap solver on the coarse time grid and an expensive solver on the fine time grids. The solution within each coarse interval is propagated independently using the fine solver. The mismatch of the solution between the coarse solution and fine solution is corrected iteratively. The theoretical speedup can be achieved is the ratio of the coarse interval number and iteration number. In this dissertation, the Parareal algorithm is tested on the North American eastern interconnection system with around 70,000 buses and 5,000 generators.

Keywords: Adomian decomposition method; parallel computing; power system simulation; semi-analytical solution; transient stability; High-performance computing, parallel algorithms, Parareal-in-time.

TABLE OF CONTENTS

Chapter One Introduction and Background Information 1

1.1 Introduction 1

1.2 Semi-analytical approaches 3

1.2.1 ADM 4

1.2.2 VIM 4

1.2.3 HAM 5

1.3 Existing Parallel Computing Algorithms 5

1.3.1 Parareal 6

1.3.2 PITA 6

1.3.3 Waveform relaxation 7

1.3.4 Spatial decomposition 7

Chapter Two SAS Based Deterministic Power System Simulation 9

2.1 Solving Power System DEs Using the ADM 9

2.1.1 Adomian Decomposition Method 9

2.1.2 Deriving an ADM-based SAS of Power System DEs 11

2.1.3 Adomian Convergence and Time Window of Accuracy for an SAS 15

2.1.4 Evaluating an SAS Using an Adaptive Time Window 23

2.2 SAS-based Scheme for Power System Simulation 25

2.2.1 Offline Stage 25

2.2.2 Online Stage 27

2.3 Case Studies on the IEEE 39-bus System 29

2.3.1 Fixed Time Window 29

2.3.2 Adaptive Time Window.. 38

2.3.3 Time Performance.. 46

2.3.4 Simulation of a contingency with multiple disturbances................................ 52

2.3.5 Simulating Systems with Non-conforming Load... 57

Chapter Three SAS Based Stochastic Power System Simulation 61

3.1 Stochastic Simulation Approaches 61

3.2 Power System SDE Model with Stochastic Loads 64

3.2.1 Synchronous Generator Modeling 64

3.2.2 Stochastic Load Modeling 64

3.3 Proposed ADM-based Approach for Solving Power System SDEs...................... 66

3.3.1 Modeling Stochastic Variables 66

3.3.2 Solving SDEs Using the ADM 66

3.4 Comparison between the Euler-Maruyama Approach and ADM-based Approach 68

3.5 Stability of Stochastic Systems.. 73

3.6 Case Studies.. 74

3.6.1 Stochastic Loads at 5% with Low Variances.. 74

3.6.2 Stochastic Loads at 100% with Low Variances.. 76

3.6.3 Stochastic Loads at 100% with High Variances 78

3.6.4 Variances of State Variables.. 82

3.6.5 Comparison on Time Performance 84

3.7 Conclusion 84

Chapter Four Parareal in Time Power System Simulation 86

4.1 Parareal Application to Power System 86

4.2 Simplified Generator Model 90

4.3 Simplified Generator Model Case Study 93

4.3.1 IEEE 3-generator 9-bus system.. 94

4.3.2 IEEE New England 10-generator 39-bus system.. 97

4.3.3 IEEE 327-generator 2382-bus Polish system.. 99

4.4 Embedding Spatial Decomposition in Parareal in Time Algorithm 101

4.4.1 Initial Coarse Propagation.. 103

4.4.2 Fine Propagation 105

4.4.3 Coarse Propagation 106

4.5 Case Study for Spatial Temporal Decomposition.. 107

4.6 Test on EI system... 114

4.6.1 EI System Overview 114

4.6.2 Solving the Network Equation.. 115

4.6.3 Test Results.. 118

4.7 Improving the Convergence Rate of Parareal-in-time Power System Simulation using the Krylov Subspace 120

4.7.1 Projection to the Krylov Subspace... 121

4.7.2 Construction of Krylov Subspace and Its Basis.. 124

4.7.3 Computational Structure 124

4.7.4 Relation between the Growth of the Krylov Subspace and Convergence Rate 126

4.7.5 Case Study 131

Chapter Five Conclusions and Future Work.. 133

List of References 134

Appendix... 144

Publications.. 145

Vita.. 147

LIST OF TABLES

Table 2.1. Parameters of the SMIB system. 16

Table 2.2. $T_{\max}$ vs. Time Constants of the System. 19

Table 2.3. The Number of CUs for the 4th-Order Model System. 47

Table 2.4. The Number of CUs for the 6th-Order Model System. 48

Table 2.5. Time Performance on the 4th-Order model System. 50

Table 2.6. Time Performance on the 6th-Order model System. 50

Table 2.7. Influence of Parallel Capability on Time Performance. 51

Table 3.1. Time performance on the 4th-Order model System. 84

Table 4.1. IEEE 3-Generator 9-Bus System Time Performance. 95

Table 4.2. IEEE 10-Generator 39-Bus System Time Performance. 97

Table 4.3. IEEE 327-Generator 2383-Bus Polish System Time Performance. 99

Table 4.4. Time Performance of Spatial Temporal Decomposition Parareal. 110

Table 4.5. EI system statistics. 115

Table 4.6. Numbers of iterations. 129

LIST OF FIGURES

Fig. 2.1. Comparison of SASs with numerical result. 17

Fig. 2.2. Different terms of the SAS and the time window of accuracy. 18

Fig. 2.3. Relationships between $T_{\max}$, T_2, T_1 and H_3. 20

Fig. 2.4. Tmax's with respect to selected $H3$'s. 20

Fig. 2.5. Using an initial state with ω(0)=0 rad/s and δ(0)=0.76 rad. 21

Fig. 2.6. Using an initial state with $\omega(0)$=1.38 rad/s and $\delta(0)$=0.04 rad. 21

Fig. 2.7. Flowchart of the proposed approach. 26

Fig. 2.8. IEEE 10-generator 39-bus system. 29

Fig. 2.9. Comparison of the simulation results given by the R-K 4 and the 2-term SAS using a fixed time window of 0.001 s. 30

Fig. 2.10. Comparison of the simulation results given by the R-K 4 and the 2-term SAS using a fixed time window of 0.01 s. 34

Fig. 2.11. Comparison of the simulation results of rotor speeds given by the R-K 4 and the 2-term SAS using a fixed time window of 0.02 s. 37

Fig. 2.12. Comparison of the simulations using the 6th-order generator model by the R-K 4 and the 2-term SAS using a fixed time window of 0.001 s. 39

Fig. 2.14. Comparison of rotor angles given by the R-K 4 and the 3-term SAS using an adaptive time window initiated from 0.001 s. 43

Fig. 2.13. Estimation of $I_{D,\max}$ 44

Fig. 2.15. Adaptive changing of time window length. 45

Fig. 2.16. Comparison of the simulation results with a topology change at t=3 s given by the R-K 4 and a 3-term SAS using an adaptive time window. 53

Fig. 2.17. The IEEE 10-machine 39-bus system with non-conforming load rotor speed deviation simulation comparison between the R-K 4 (blue) and SAS based approach (red). 60

Fig. 2.18. The changing of time window length for the IEEE 10-machine 39-bus system with non-conforming load simulation. 60

Fig. 3.1. Stochastic modeling approaches. 62

Fig. 3.2. SMIB system with constant impedance load at generator bus. 68

Fig. 3.3. Simulation results of generator 1 rotor angle with loads connecting to bus 3 and 4 represented by stochastic variable with 2% load variation. 75

Fig. 3.4. Simulation results of generator 1 rotor angle with all loads represented by stochastic variable with 2% load variation. 77

Fig. 3.5. Simulation results of generator 1 rotor angle with all loads represented by stochastic variable with 4% load variation. 79

Fig. 3.6. Simulation results of bus voltage at bus 30 with all loads represented by stochastic variable with 4% load variation. 80

Fig. 3.7. Evolution of the pdf of the voltage magnitude at bus 30 from t=10 s to 20 s. ... 81

Fig. 3.8. Mean value of generator 1's rotor angle for case A. 82

Fig. 3.9. Standard deviation of generator 1's rotor angle for case A. 82

Fig. 3.10. Mean value of generator 1's rotor angle for case B. 83

Fig. 3.11. Standard deviation of generator 1's rotor angle for case B. 83

Fig. 4.1. 2.2 model of a synchronous machine. 91

Fig. 4.2. Initial coarse solution comparison of 2.2 and 1.1 model.................................... 94

Fig. 4.3. Iterations of 1st window with 1.1 model. 96

Fig. 4.4. Iterations of 10th window with 1.1 model. 98

Fig. 4.5. Iterations of 10th window with 1.1 model. 100

Fig. 4.6. Spatial decomposition that decomposes the whole system into 2 subsystems. 102

Fig. 4.7. Spatial decomposition simulation procedure. 104

Fig. 4.8. Test system. 108

Fig. 4.9. Comparison of relative rotor angles (machine 13 in area 1 as reference) in area 1 from sequential simulation (blue curves) and spatial-temporal decomposition (red curves). 109

Fig. 4.10. Different steps and their percentage in the total computing time.................. 111

Fig. 4.11. Compass plot of rotor angle terms of the inter-area mode eigenvector.......... 112

Fig. 4.12. Comparison of relative rotor angles (machine 13 in area 1 as reference) of machines 4, 6, 7 and 9 in area 1 and machine 3 in area 2 from sequential simulation (blue curves) and from spatial-temporal decomposition (red curves). . 113

Fig. 4.13. Non-zero elements in the EI system **Y** matrix.. 116

Fig. 4.14. LU factorization of EI system **Y** matrix. 116

Fig. 4.15. Non-zero elements in the reordered EI system **Y** matrix. 117

Fig. 4.16. LU factorization of the reordered EI system **Y** matrix. 117

Fig. 4.17. GUI for Parareal Algorithm. 118

Fig. 4.18. Iteration numbers of the Parareal algorithm. 119

Fig. 4.19. Selected simulation results of the fault bus and a neighboring bus voltage... 120

Fig. 4.20. Computational structure of parareal algorithm with convergence accelerator. 125

Fig. 4.21.Rotor angle simulation result from the Krylov subspace enhanced parareal approach. 126

Fig. 4.22. Krylov subspace's basis for different iterations of the damped SMIB system simulation... 127

Fig. 4.23. Error comparison between the Krylov subspace-enhanced parareal and the plain parareal for each iteration (damped SMIB system). 128

Fig. 4.24. Rotor angle simulation result from the Krylov subspace-enhanced parareal approach for the undamped system under large disturbance 129

Fig. 4.25. Contingency screening procedure having the options of both plain parareal and the Krylov subspace-enhanced parareal.. 130

Fig. 4.26. IEEE 10-machine 39-bus system rotor speed deviation simulation results from the Krylov subspace-enhanced parareal approach. 131

Fig. 4.27. Error comparison between the Krylov subspace-enhanced parareal and the plain parareal for each iteration (IEEE 10-machine 39-bus system). 132

CHAPTER ONE

INTRODUCTION AND BACKGROUND INFORMATION

1.1 Introduction

Power system stability assessment is a challenging topic due to the complexity of a modern power grid and the wide range of the time constants of devices. In daily operations, power system dynamic security assessment needs to be performed on a regular basis, which involves assessing the angular stability and voltage stability to predict possible instability and provide insights for control actions. Beyond the dynamic security assessment, steady-state security of the system involving severe contingencies such as cascading failure [1][2] also needs to be studied although not as frequently as the dynamic security assessment.

Voltage stability, to its nature, is a local stability problem. The mitigation of voltage instability requires local reactive power support. In the procedure of deciding optimal locations [3] and sizes [4] of reactive power supports, time domain simulation plays an important role, the effectiveness of the reactive power supports need to be validated through time domain simulation. In system planning studies, hundreds of contingencies need to be simulated. The optimization algorithm also requires several iterations to find the optimal decision. This means a large number of simulations must be performed. If the time performance of power system simulation can be improved, it will enable the planning engineers to perform such studies more frequently.

On the other hand, angular stability problems involve faster dynamics than voltage stability problems and are not necessarily local. There are two ways of assessing angular stability problems, analytical and numerical. Both have advantages and drawbacks. An Analytical method is computationally inexpensive; it evaluates the system angular stability by an energy function-based direct method which may only need the fault-on simulation. However, most of the analytical methods are over conservative, and some of them can not fully capture the nonlinearity of a power system. Some studies have investigated the relation between the amplitude and the frequency of oscillation analytically [5][6]. There are also studies introducing a measurement-based approach for identifying the oscillation mode [7]. On the other hand, numerical methods simulate the step-by-step response of the system, which provides more accurate information about the system. But for systems with realistic models and sizes, numerical methods are computationally demanding, which prohibit them from being utilized in real-time security assessment applications.

Time-domain simulation of a power system following a contingency for transient stability analysis needs to solve nonlinear differential-algebraic equations (DAEs) on the system state over a simulation period. Numerical integration methods, either explicit or implicit, are traditionally employed to solve the Initial Value Problem (IVP) of the DAEs but their iterative computations could be time-consuming for a multi-machine power system because the DAEs in nature model tight coupling between machines via nonlinear sine functions. A very small integration step, typically less than one millisecond, is usually required for accuracy of the integration. Thus, a large number of iterations are needed at each step, but numerical instability may become another concern with explicit integration methods like Runge–Kutta methods, which is widely applied in today's simulation software. Implicit integration methods like the Trapezoidal method overcome numerical instability by introducing implicit algebraic equations, which also need to be solved thru iterations by numerical methods like the Newton-Raphson method, and thus, the computational complexity is significantly increased.

1.2 Semi-analytical approaches

The difficulty of solving a nonlinear ordinary differential equation (ODE) is mainly caused by its nonlinear expression. In traditional approaches, if an analytical expression is desired, linearization is applied to nonlinear functions to facilitate analytic approaches such as Laplace Transform at the cost of nonlinearities. On the other hand, if the system's nonlinear time-domain response following a contingency is of interest, numerical approach is applied. There have been studies trying to combine the advantages of the analytic approach and numerical approach, namely, yielding an approximate analytical solution which is accurate within certain time range and keeps the system nonlinearity intact at the same time. This type of methods is called "semi-analytical".

There are three dominant sub-categories of semi-analytical methods, the Adomian Decomposition Method (ADM), the Homotopy Analysis Method (HAM) and the Variational Iteration Method (VIM). The ADM is the developed by George Adomian from the 1970s to the 1990s, the HAM is proposed by Shijun Liao in 1992 and the VIM is proposed by Ji-Huan He in 1999. Among them, the ADM is considered the foundation of decomposition methods. Both the HAM and the VIM use the ADM as a benchmark to test their performances [8][9].

1.2.1 ADM

The ADM allows the system solution to be expressed in terms of a nonlinear expressions without linearize the system. The essential technique of the ADM is to recursively develop a series of Adomian polynomials which are generalized Maclaurin series to approximately represent the nonlinear functions in the system differential equations.

In [10], Abbaoui and Cherruault provided a new idea of proving the convergence of the ADM. They proved the convergence of Adomian polynomials is equivalent to the convergence of the approximate solution. To prove the convergence of Adomian polynomials, they first redefined the way of calculating them [11], and then used number theory to find the convergence range. Using the convergence range found in [10], an efficient step-size control method is proposed in [12] to improve the speed of the ADM.

1.2.2 VIM

The key idea of the VIM is to use the iteration of a correction function to acquire several successive approximations of an exact solution. However, instead of summing up all of them as in the ADM and the HAM, the exact solution is obtained at the limit of these aforementioned approximations [1]. Because the ADM provides the components of an approximate solution, each of those components can be utilized for the analysis of system. On the other hand, the only useful result of the VIM is the last approximation given by iteration, the results before the final iteration are discarded. However, because the VIM reduces the volume of calculations by not requiring the Adomian polynomials, it gives the solution more rapidly comparing to the ADM.

1.2.3 HAM

In [9], the ADM is compared with the HAM and its special case the Homotopy Perturbation Method (HPM). Both the ADM and the HPM are methods which consider the approximate solution of a nonlinear equation as an infinite series usually converging to the accurate solution [13]. Therefore the ADM and the HPM are equivalent in solving nonlinear equations [13] and they are special cases of the HAM. The advantage of the HAM compared to the ADM and the HPM is that the solution can be expressed by different base functions rather than polynomial functions [9].

1.3 Existing Parallel Computing Algorithms

Although sequential numerical integration approaches are still the first choices of most simulation software, the power industry has expressed interests in alternative approaches that can improve the time performance of power system simulation. Power systems have two characteristics that limit the speed of simulation: the tight coupling between state variables and the sequential nature of time evolution. Any parallel simulation approach has to address those two characteristics in order to achieve a better performance than sequential numerical integration approaches.

1.3.1 Parareal

The Parareal-in-time method is a variation of the multi-shooting method [14]. It decomposes the time domain into sub-intervals and uses a computationally cheap (coarse) solver to globally propagate a less accurate solution. Then within each sub-interval, an accurate (fine) solver is utilized to correct the solution acquired from the coarse solver to the true solution.

The Parareal method is designed for parallel computing. If enough parallel processors are available, the correction of each sub-interval can be computed concurrently by different processors. Under an ideal parallelism, the time for computing the entire true solution trajectory will be the time for correcting one sub-interval's coarse solution to the true solution, multiplied by the number of required iterations.

1.3.2 PITA

The Parallel implicit time-integrator (PITA) is closely related to the Parareal-in-time algorithm [15]. The difference is that the PITA has a different coarse solution correction scheme. The difference between coarse solution and fine solution at each coarse time instant is called a jump in the PITA. Since the jumps are propagated using the same set of DEs as the state variables, it is observed that those jumps oscillate at a comparable frequency with the system's nature frequency. To eliminate the numerical resonance caused by the propagation of the jumps, the PITA uses the solution from previous iterations to construct a Krylov subspace and project the correction to this subspace, therefore filter out the nature modes from the jumps.

1.3.3 Waveform relaxation

Instead of designing parallel computation in the dimension of time, waveform relaxation considers the parallelism across different state variables. Although dealing with different types of parallelism, the first step of the waveform relaxation approach is similar to Parareal. It also involves an initial guess of the solution (or waveform) of state variables. The state variables are separated into several groups based on offline studies of the system. The state variables between groups are assumed to be coupled not as tight as those within one group. For each group of state variables, the simulation is carried out independent with any solution of state variables belonging to other groups fixed to a presumed waveform. Then the solution of the entire system is corrected towards the true solution iteratively.

Paper [16] applies waveform relaxation approach to implement parallelism through state variables by assuming and correcting waveforms for them. The grouping of state variables is based on the geographic location of the machines and the initial guesses of states variables outside the interested group are assumed to be constant values during the numerical integration of the state variables inside the interested group.

1.3.4 Spatial decomposition

In the power simulation studies, the spatial decomposition approach is a more intuitive way of designing parallel computing across different areas. Paper [17] decomposes the system into a study area and the external area. For each time step, the simulation of each area is independent. The information is passed between areas using fictitious generators at the boundary buses. During each time step, voltages of the boundary buses are solved through balancing the power flow equations taking the power inputs from both sides of the boundary.

CHAPTER TWO

SAS BASED DETERMINISTIC POWER SYSTEM SIMULATION

Intuitively, to solve a power system's DEs, if the analytical solution of the IVP about each state variable could be found as an explicit, closed-form function about symbolic variables including time, the initial state and other variables on the system operating condition, such a function would directly give the state value at any time instant without conducting time-consuming computations or iterations through all integration steps as R-K 4 does. However, for nonlinear power system DEs, such an analytical solution being accurate for any simulation time period does not exist in theory. Thus, a compromise is to find an approximate analytic solution, named a semi-analytical solution (SAS), which keeps accuracy for a certain length of time window (denoted by *T*), and can be repeatedly used over a series of such windows until those windows make up a desired simulation period. If an SAS is derived beforehand, then solving the IVP becomes simply evaluating the SAS, i.e. plugging in values of symbolic variables, which can be extremely fast compared to numerical integration. If online evaluation of the SAS for each window *T* takes a short computation time τ, the T/τ indicates how many times the SAS-based power system simulation can be faster than the wall-clock time.

2.1 Solving Power System DEs Using the ADM

2.1.1 Adomian Decomposition Method

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) \tag{2-1}$$

$$\mathbf{x}(t) = \left[x_1(t) \quad x_2(t) \quad \dots \quad x_M(t)\right]^T$$

$$\mathbf{f}(\cdot) = \left[f_1(\cdot) \quad f_2(\cdot) \quad \dots \quad f_M(\cdot)\right]^T$$

Consider a nonlinear dynamic system, e.g., a power system, with M state variables modeled by nonlinear DE (2-1).

$$\mathscr{L}[\mathbf{x}] = \frac{\mathbf{x}(0)}{s} + \frac{\mathscr{L}[\mathbf{f}(\mathbf{x})]}{s} \tag{2-2}$$

To solve **x**(*t*), the first step of the ADM is to apply Laplace transform $\mathscr{L}[\cdot]$ to transform (2-1) into an algebraic equation (AE) about complex frequency *s* [18], [19], and then solve $\mathscr{L}[\mathbf{x}]$ to obtain (2-2).

$$\mathbf{x}(t) = \sum_{n=0}^{\infty} \mathbf{x}_n(t) \tag{2-3}$$

$$f_i(\mathbf{x}) = \sum_{n=0}^{\infty} A_{i,n}(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_n), \quad i = 1 \cdots M \tag{2-4}$$

$$A_{i,n} = \frac{1}{n!}\left[\frac{\partial^n}{\partial \lambda^n} f_i\left(\sum_{i=0}^{n} \mathbf{x}_i \lambda^i\right)\right]\Bigg|_{\lambda=0} \tag{2-5}$$

Assume that **x(***t***)** can be decomposed as (2-3). Then, use (2-4) to decompose each $f_i(\cdot)$, i.e. **f(·)**'s *i*-th element, as a sum of infinite Adomian polynomials given by (2-5), where *λ* is called a grouping factor [20].

$$\mathscr{L}\left[\mathbf{x}_0\right] = \mathbf{x}(0)/s \tag{2-6}$$

$$\mathscr{L}\left[\mathbf{x}_{n+1}\right] = \mathscr{L}\left[\mathbf{A}_n\right]/s \qquad n \geq 0 \tag{2-7}$$

Matching the terms of **x**(*t*) and **f**(·) with the same index [21], we can easily derive recursive formulas (2-6) and (2-7) for L [$\mathbf{x}_n$] ($n\geq 0$), where $\mathbf{A}_n=[\,A_{1,n}\ ,\ \ldots\ A_{M,n}]^{\mathrm{T}}$.

By applying an inverse Laplace transform $\mathscr{L}^{-1}[\cdot]$ to both sides of (2-6) and (2-7), we can obtain $\mathbf{x}_n(t)$ for any *n*. An SAS of (2-1) is defined as the sum of first *N* terms of $\mathbf{x}_n(t)$:

$$\mathbf{x}^{SAS}\left(t\right)=\sum_{n=0}^{N-1}\mathbf{x}_{n}\left(t\right) \qquad \text{(2-8)}$$

2.1.2 Deriving an ADM-based SAS of Power System DEs

$$\begin{cases}\dot{\delta}_k=\omega_k-\omega_R\\ \dot{\omega}_k=\dfrac{\omega_R}{2H_k}(P_{mk}-P_{ek}-D_k\dfrac{\omega_k-\omega_R}{\omega_R})\\ \dot{e}'_{qk}=\dfrac{1}{T'_{d0k}}\left[E_{fdk}-e'_{qk}-(x_{dk}-x'_{dk})i_{dk}\right]\\ \dot{e}'_{dk}=\dfrac{1}{T'_{q0k}}\left[-e'_{dk}+(x_{qk}-x'_{qk})i_{qk}\right]\end{cases} \qquad \text{(2-9)}$$

$$\begin{aligned}&E_k=e'_{dk}\sin\delta_k+e'_{qk}\cos\delta_k+j(e'_{qk}\sin\delta_k-e'_{dk}\cos\delta_k)\\ &I_{tk}=i_{Rk}+ji_{Ik}\overset{\text{def}}{=}\mathbf{Y}_k^{*}\mathbf{E}\\ &P_{ek}=e_{qk}i_{qk}+e_{dk}i_{dk}\\ &i_{qk}=i_{Ik}\sin\delta_k+i_{Rk}\cos\delta_k,\quad i_{dk}=i_{Rk}\sin\delta_k-i_{Ik}\cos\delta_k\\ &e_{qk}=e'_{qk}-x'_{dk}i_{dk},\qquad\qquad e_{dk}=e'_{dk}+x'_{qk}i_{qk}\\ &V_{k=}\sqrt{e_{dk}^2+e_{qk}^2}\end{aligned} \qquad \text{(2-10)}$$

For a power system having K synchronous generators, consider the 4th-order two-axis model (2-9) to model each generator with saliency ignored [22]. All generators are coupled through nonlinear AEs in (2-10) about the network. In (2-9) and (2-10), ω_R is the rated angular frequency; δ_k , ω_k, H_k and D_k are respectively the rotor angle, rotor speed, inertia and damping coefficient of the machine k; $\mathbf{Y}_k$ is the kth row of the reduced admittance matrix $\mathbf{Y}$; $\mathbf{E}$ is the column vector of all generator's electromotive forces (EMFs) and E_k is the kth element; P_{mk} and P_{ek} are the mechanical and electric powers; E_{fdk} is the internal field voltage; e'_{qk}, e'_{dk}, i_{qk}, i_{dk}, T'_{q0k}, T'_{d0k}, x_{qk}, x_{dk}, x'_{qk} and x'_{dk} are transient voltages, stator currents, open-circuit time constants, synchronous reactances and transient reactances in q- and d-axes, respectively; V_k is the terminal bus voltage magnitude.

In addition, consider the following first-order exciter and governor models [23]:

$$\dot{E}_{fdk} = \frac{1}{T_{Ak}}\left[-E_{fdk} + K_{Ak}(V_{refk} - V_k)\right] \tag{2-11}$$

$$\dot{P}_{mk} = \frac{1}{T_{gk}}(-P_{mk} + P_{refk} - \frac{\omega_k - \omega_R}{R_k}) \tag{2-12}$$

where T_{Ak} and K_{Ak} are respectively the time constant and gain in voltage regulation with the exciter, V_{refk} is the reference voltage value, T_{gk} is total time constant of the governor and turbine, P_{refk} is the setting point of the mechanical power output, R_k is the speed regulation factor.

In the following context, the 4th-order model is utilized as an example to illustrate the derivation of an SAS for simplicity of description. A similar procedure is applied to the 6th-order DE model in (2-9), (2-11) and (2-12) and other DE models. Substitute AEs (2-10) into DEs (2-9) to eliminate i_{qk}, i_{dk} and P_{ek}. Then, the differential-algebraic equations (2-9) and (2-10) are transformed into the form of (2-1), where state vector $\mathbf{x}=\begin{bmatrix}\delta_1 & \omega_1 & e'_{q1} & e'_{d1} & \dots & \delta_K & \omega_K & e'_{qK} & e'_{dK}\end{bmatrix}^T$ has M=4K state variables as the elements. Then, an SAS of this set of DEs can be derived by formulas (2-6) and (2-7), as illustrated below about the generator speed ω of a single-machine infinite-bus (SMIB) system modeled by (2-9). Assume that the infinite bus has voltage V_∞=1 pu. Let $\mathbf{x}=[\delta,\omega,e'_q,e'_d]^T$ and $\mathbf{f}=[f_1,f_2,f_3,f_4]^T$, which are the nonlinear functions in four DEs. From (2-3),

$$\delta(t)=\sum\nolimits_{n=0}^{\infty}\delta_n(t) \tag{2-13a}$$

$$\omega(t)=\sum\nolimits_{n=0}^{\infty}\omega_n(t) \tag{2-13b}$$

$$e'_d(t)=\sum\nolimits_{n=0}^{\infty}e'_{d,n}(t) \tag{2-13c}$$

$$e'_q(t)=\sum\nolimits_{n=0}^{\infty}e'_{q,n}(t) \tag{2-13d}$$

Then, equation (2-2) about ω becomes

$$\mathscr{L}[\omega]=\frac{\omega(0)}{s}+\frac{\mathscr{L}[f_2(\delta,\omega,e'_q,e'_d)]}{s} \tag{2-14}$$

From (2-4) and (2-5), the first two Adomian polynomials for f_2 are given in

$$A_{2,0}=\frac{\omega_R}{2H}\left[-e'_{q,0}\gamma_1-e'_{d,0}\gamma_2+\gamma_1\gamma_2(x'_d-x'_q)+P_m-D\frac{\omega_0-\omega_R}{\omega_R}\right] \tag{2-15a}$$

$$
\begin{aligned}
A_{2,1} = \frac{\omega_R}{2H}[&Y_o x'_d (e'_{q,1}\gamma_2 + e'_{d,1}\gamma_1) - \delta_1 Y_\infty^2 (x'_d - x'_q)\cos 2\delta_0 \\
&- 2Y_\infty (e'_{q,0}e'_{q,1} + e'_{d,0}e'_{d,1}) - Y_\infty (e'_{d,1}\cos\delta_0 - e'_{q,1}\sin\delta_0) \\
&- x'_d Y_o \gamma_4 + Y_o Y_\infty \delta_1 x'_q \gamma_4 + Y_\infty \delta_1 \gamma_3 - Y_o x'_q (e'_{d,1}\gamma_1 + e'_{q,1}\gamma_2) - \frac{D\omega_1}{\omega_R}]
\end{aligned}
\qquad (2\text{-}15b)
$$

where

$$\gamma_1 = Y_o e'_{q,0} - Y_\infty \sin\delta_0$$

$$\gamma_2 = Y_o e'_{d,0} + Y_\infty \cos\delta_0$$

$$\gamma_3 = e'_{q,0}\cos\delta_0 + e'_{d,0}\sin\delta_0$$

$$\gamma_4 = e'_{d,0}\cos\delta_0 + e'_{q,0}\sin\delta_0$$

where Y_o and $Y_\infty=|Y_\infty|\angle\beta$ are respectively the admittances from the generator's EMF to the ground and to the infinite bus. Note that E_{fd}, which is constant in this 4th-order DE model, only explicitly appears in the Adomian polynomials about e'_q.

Since the accuracy of an SAS defined by (2-8) only lasts for a limited time window T [24], [25], [26], [27], a multi-stage strategy, i.e. the M-ADM [28], [29], [30], [31], is adopted to extend the accuracy of the same SAS to an expected simulation period by these two steps:

Step-1: Partition the simulation period into sequential windows of T each able to keep an acceptable accuracy of the SAS.

Step-2: Evaluate the SAS at desired time points in the first T using the given initial state and the values of other symbolic variables; starting from the second window T, evaluate the SAS by taking the final state of the previous T as the initial state.

As long as the final state of each window is accurate enough, the accuracy of the next window will be ensured. To apply this approach to simulate a contingency, we may first perform the numerical approach until the contingency is cleared to obtain the initial state for the IVP about the post-contingency simulation period, and then the M-ADM can be performed.

2.1.3 Adomian Convergence and Time Window of Accuracy for an SAS

This subsection studies the convergence and time window of accuracy of the ADM-based SAS. First, consider an SMIB system having a 2nd-order classical model generator connected to the infinite bus by an impedance. Thus, Y_o is zero and the EMF E of the generator has a constant magnitude so as to eliminate two DEs on e'_d and e'_q in (2-9). System parameters and initial conditions are listed in Table 2.1. Mechanical power P_m determines the operating condition. V_∞ is the voltage magnitude of the infinite bus, whose phase angle is considered zero. $\delta(0)$ and $\omega(0)$ are the initial rotor angle and speed of the generator, which are initial state variables.

$$\begin{aligned}\delta(t)=\sum_{n=0}^{4}\delta_n &=11.39t^8+4.50t^7+1353.32t^6+361.02t^5\\ &\quad -240.09t^4-47.72t^3+20.81t^2+2.05t+0.06\end{aligned} \tag{2-16}$$

where

$\delta_0=0.06$, $\delta_1 = 20.81t^2 + 2.05t$, $\delta_2=-241.61t^4-47.72t^3$

$\delta_3=1184.67t^6+350.95t^5+1.52t^4$, $\delta_4=11.39t^8+4.50t^7+168.66t^6+10.07t^5$

Table 2.1. Parameters of the SMIB system.

Parameter	Value
H	3 s
D	0 s
$Y_\infty = \lvert Y_\infty \rvert \angle \beta$	$0.9\angle 90°$pu
Y_o	0 pu
P_m	0.8 pu
$\lvert E \rvert$	1.1 pu
V_∞	1
ω_R	377 rad/s
$\delta(0)$	0.06 rad
$\omega(0)$	2.05 rad/s

For N=5, 5 terms of the SAS are given in (2-16) as an example and its trajectory and the trajectories of individual terms are shown in Fig. 2.2.

Fig. 2.1 plots the trajectories of six different SASs with N=3 to 8, respectively, and compare them with the numerical integration result from the R-K 4.

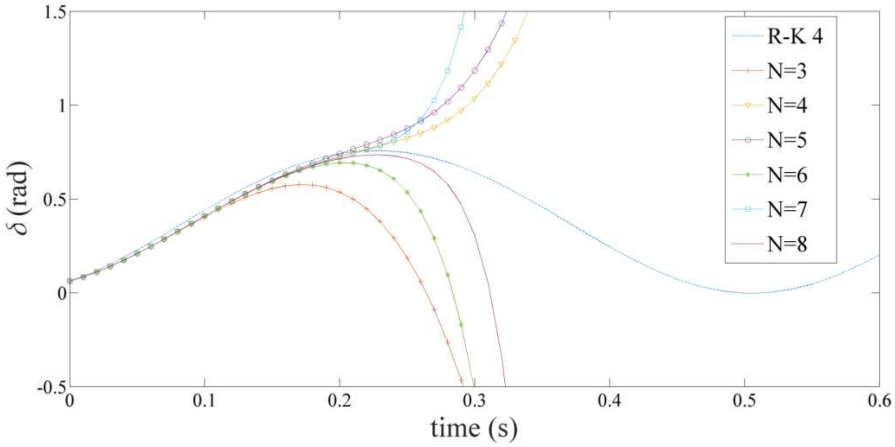


Fig. 2.1. Comparison of SASs with numerical result.

In Fig. 2.2, $T_{\max}$ denotes a limit of the time window of accuracy. Also define the absolute value of the last term, i.e.$|x_{N\text{-}1}|$, as a divergence indicator I_D, which is close to zero within $T_{\max}$ and sharply increases the magnitude, otherwise. $T_{\max}$ can be estimated by selecting an appropriate threshold $I_{D,\max}$ for I_D. For instance in Fig. 2.2, $I_{D,\max}$ is set at 0.01 rad to determine $T_{\max}$. There are two observations from Fig. 2.2:

- The SAS from the ADM matches well the R-K 4 result within 0.2s, i.e. a time window of accuracy.
- The higher order of a term, the less contribution it has and the faster it diverges to infinity. The last term δ_4 diverges quickly outside 0.2s.

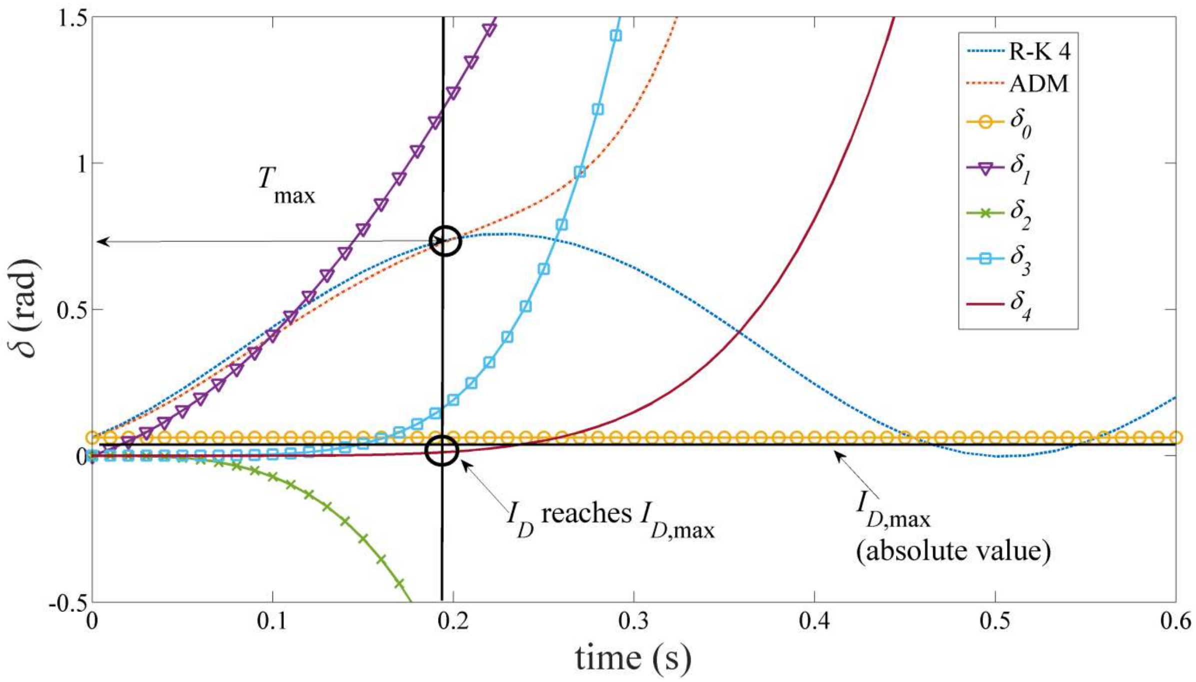


Fig. 2.2. Different terms of the SAS and the time window of accuracy.

To unveil the relation between $T_{\max}$ and time constants of a multi-machine system, the IEEE 3-generator 9-bus system in [32] is studied. Gradually decrease H_3, the inertia of generator 3, from 4.5 s to 1.0 s while keeping the other two unchanged at original 23.64 s and 6.4 s, such that eight system models are yielded as shown in Table 2.2. Because the system has two oscillation modes and their oscillation periods T_1 and T_2 may be important time constants influencing $T_{\max}$, T_1 and T_2 are estimated from each linearized model of the system and are listed in Table 2.2. A three-phase fault at bus 7 cleared by tripping line 5-7 is simulated on each model by both the R-K 4 and the ADM with N=3 (using the post-fault state from the R-K 4 as its initial state). Using 0.01 rad as $I_{D,\max}$, the estimated $T_{\max}$ for each model is given in the table

Table 2.2. $T_{\max}$ vs. Time Constants of the System.

No.	H_3(s)	T_1(s)	T_2(s)	$T_{\max}$(s)
1	4.5	0.9510	0.5516	0.2546
2	4.0	0.9438	0.5280	0.2342
3	3.5	0.9369	0.5014	0.2131
4	3.0	0.9304	0.4718	0.1905
5	2.5	0.9241	0.4365	0.1662
6	2.0	0.9183	0.3961	0.1410
7	1.5	0.9128	0.3479	0.1137
8	1.0	0.9076	0.2881	0.0845

. Fig. 2.3 illustrates that T_1, T_2 and $T_{\max}$ monotonically increase with H_3. The bigger time constant T_1 does not change significantly with H_3.

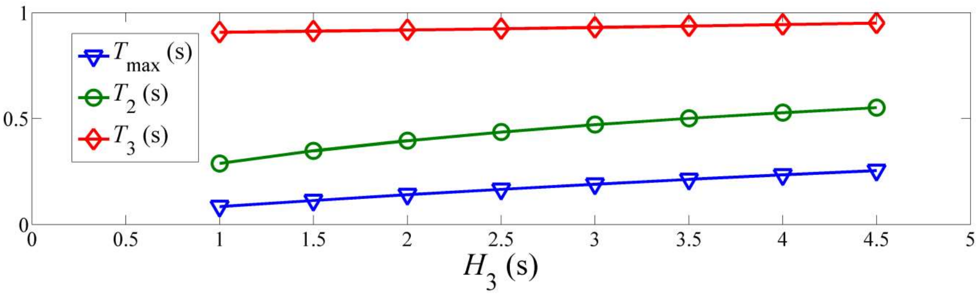


Fig. 2.3. Relationships between $T_{\max}$, T_2, T_1 and H_3.

Fig. 2.4 shows values of $T_{\max}$ for H_3=1.5 s, 3 s and 4.5 s, beyond which the ADM result starts diverging from the R-K 4 result. A hypothesis for a multi-machine power system is that $T_{\max}$ is mainly influenced by the smallest time constant.

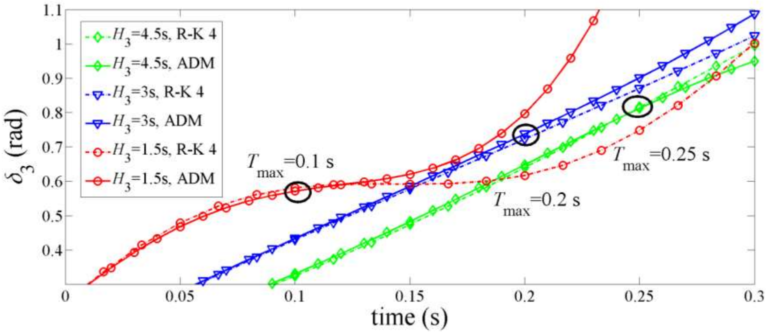


Fig. 2.4. $T_{\max}$'s with respect to selected H_3's.

If the initial state varies, the time of accuracy may change as well. For the above SMIB system, different values of $\delta(0)$ and $\omega(0)$ will lead to different $T_{\max}$'s. As illustrated by Figs. 2.5 and 2.6, the SAS evaluated starting from an initial state with $\omega(0)$=0 rad/s and $\delta(0)$=0.76 rad keeps its accuracy for a time window around 0.25s while for a larger δ (0)=1.38 rad/s and $\delta(0)$=0.04 rad, the window of accuracy may reduce to below 0.2 s.

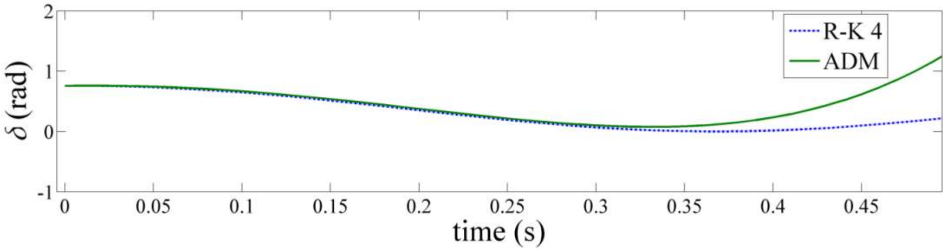


Fig. 2.5. Using an initial state with $\omega(0)$=0 rad/s and $\delta(0)$=0.76 rad.

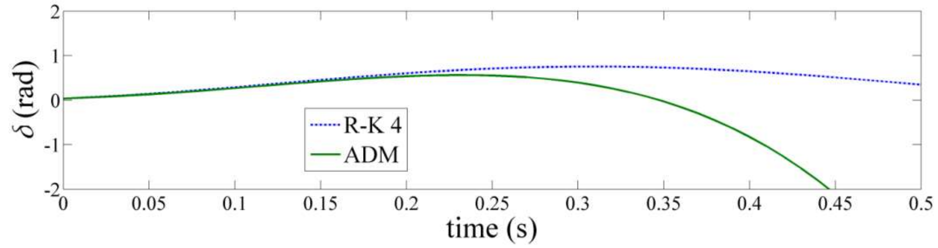


Fig. 2.6. Using an initial state with $\omega(0)$=1.38 rad/s and $\delta(0)$=0.04 rad.

For a general multi-machine system, it can be difficult to analyze how $T_{\max}$ changes with $I_{D,\max}$ about a state variable. However, we may analyze their relationship on the above SMIB system first to help gain an insight on their relationship for a multi-machine system. Consider a 3-term SAS of rotor angle δ, whose last term δ_2 has this expression

$$\delta_2 = c_1 t^4 + c_2 t^3 \tag{2-17}$$

where

$$c_1 = \frac{\omega_0^2 V_\infty \,|\, Y_\infty E \,|^2 \sin(\beta - \delta(0))}{96H^2}\left[\,|\, E \,|\cos\beta + V_\infty \cos(\beta - \delta(0)) - \frac{P_m}{|\, Y_\infty E \,|}\right]$$

$$c_2 = \frac{-\omega_0 V_\infty \,|\, Y_\infty E \,|\, (\omega(0) - \omega_R)\sin(\beta - \delta(0))}{12H}$$

Define divergence indicator I_D as δ_2 and let t=$T_{\max}$ and δ_2=$I_{D,\max}$ in (2-17) to obtain

$$I_{D,\max} = c_1 T_{\max}^4 + c_2 T_{\max}^3 \tag{2-18}$$

$T_{\max}$ has 4 roots as given in

$$T_{\max} = -\frac{c_2}{4c_1} \pm \frac{p_4}{2} \mp \frac{\sqrt{p_5 \pm p_6}}{2} \text{ or } -\frac{c_2}{4c_1} \pm \frac{p_4}{2} \pm \frac{\sqrt{p_5 \pm p_6}}{2} \tag{2-19}$$

where

$$p_1 = -27c_2^2 I_{D,\max}, \quad p_2 = p_1 + \sqrt{4 \cdot (12c_1 I_{D,\max})^3 + p_1^2}$$

$$p_3 = -\frac{4I_{D,\max}}{\sqrt[3]{2 / p_2}} + \frac{\sqrt[3]{p_2 / 2}}{3c_1} \quad , \quad p_4 = \sqrt{\frac{c_2^2}{4c_1^2} + p_3}$$

$$p_5 = \frac{c_2^2}{2c_1^2} - p_3 \quad , \quad p_6 = \frac{-c_2^3}{4c_1^3 p_4}$$

Since $T_{\max}$>0, the smallest positive root should be selected as an estimate of $T_{\max}$. For a multi-machine system, equations (2-18) and (2-19) can also be applied to approximately analyze the relationship of $I_{D,\max}$ and $T_{\max}$ for state variables of each machine by means of an SMIB equivalent about that machine against the rest of the system.

The studies above show that, for an SAS, its $T_{\max}$ depends on time constants of the system, the initial state starting the evaluation and the contingency as well. Therefore, we may either choose a fixed time window less than the most conservative $T_{\max}$ observed offline based on many simulations on probable contingency scenarios or allow the window T to change adaptively as long as divergence indicator I_D remains below a preset threshold $I_{D,\max}$ for each state variable.

2.1.4 Evaluating an SAS Using an Adaptive Time Window

The convergence of the SASs for a general nonlinear system is still an open question [33], and no sufficient condition for convergence has been proved yet. Reference [12] gives a necessary condition, i.e. the satisfaction of a ratio test: $\|\mathbf{x}_{n+1}\|_2 < \alpha \|\mathbf{x}_n\|_2$ holds for n=0, 1, …, N-1, where $0<\alpha<1$ is a constant depending on the system. However, α is difficult to derive analytically for a high-dimensional system.

This dissertation proposes a practical approach for evaluation of an ADM-based N-term SAS using an adaptive time window. The approach compares divergence indicator I_D with a preset threshold $I_{D,\max}$ to adaptively judge the end of the current window for evaluation and proceed to the next window until the entire simulation period is made up. $I_{D,\max}$ is estimated by the following procedure for a list of scenarios that each have a contingency simulated under a specific operating condition:

Step-1: For each scenario, use the post-contingency state from the R-K 4 as the initial state to run the M-ADM using a small enough fixed time window T.

Step-2: Find the maximum per unit absolute value that the last SAS term, i.e. $|x_{k,N\text{-}1}|$, of any state variable may reach over the entire simulation period. Use that value as a guess of $I_{D,\max}$.

Step-3: Add a small random variation to the post-contingency state and repeat *Step-2* for a number of times. Take the smallest guess of $I_{D,\max}$.

Step-4: After finishing *Steps 1-3* for all contingencies, choose the smallest $I_{D,\max}$ as the final threshold.

Remarks:

1) *Step-2* on guessing an $I_{D,\max}$ may exclude $\delta_{k,N\text{-}1}$, i.e. the last SAS term for each rotor angle δ_k, since its divergence can be detected through the divergence of the last SAS term of ω_k;

2) *Step-2* finds the maximum value of all last terms rather than the minimum value in order to provide a necessary condition for convergence rather than an over-conservative, sufficient condition causing loss of the advantage of using an adaptive time window;

3) The random variation in *Step-3* is added to make the $I_{D,\max}$ more independent of the post-contingency state, which may be around 1%.

The above procedure can be performed offline for potential contingencies and operating conditions. Based on our tests, $I_{D,\max}$ does not vary significantly with contingencies, so in practice, the list of scenarios does not have to be large to find an effective $I_{D,\max}$.

2.2 SAS-based Scheme for Power System Simulation

A two-stage scheme is presented for power system simulation using the M-ADM, which comprises an offline stage to derive the SASs and an online stage to evaluate the SASs as shown in Fig. 2.7.

2.2.1 Offline Stage

Assuming a constant impedance load at each bus, an SAS is derived by the ADM for each generator with symbolic variables from, e.g., one of these two groups:

- ***Group-1***: Time, the initial state, and the operating condition (e.g. generator outputs and load impedances)
- ***Group-2***: *Group-1* plus selected symbolized elements (symbolized parameters of system that subject to changes) in the system admittance matrix

Group-1 assumes a specific post-contingency system topology (i.e. a constant system admittance matrix) but relaxes the system operating condition so as to enable one SAS to simulate for multiple loading conditions. *Group-2* additionally relaxes selected elements in the admittance matrix and hence enables one SAS suitable for simulating multiple contingencies. Other symbolic variables can also be added as undetermined parameters but the more symbolic variables the more complex expression of the SAS. All SASs derived in the offline stage will be saved in storage for later online use.

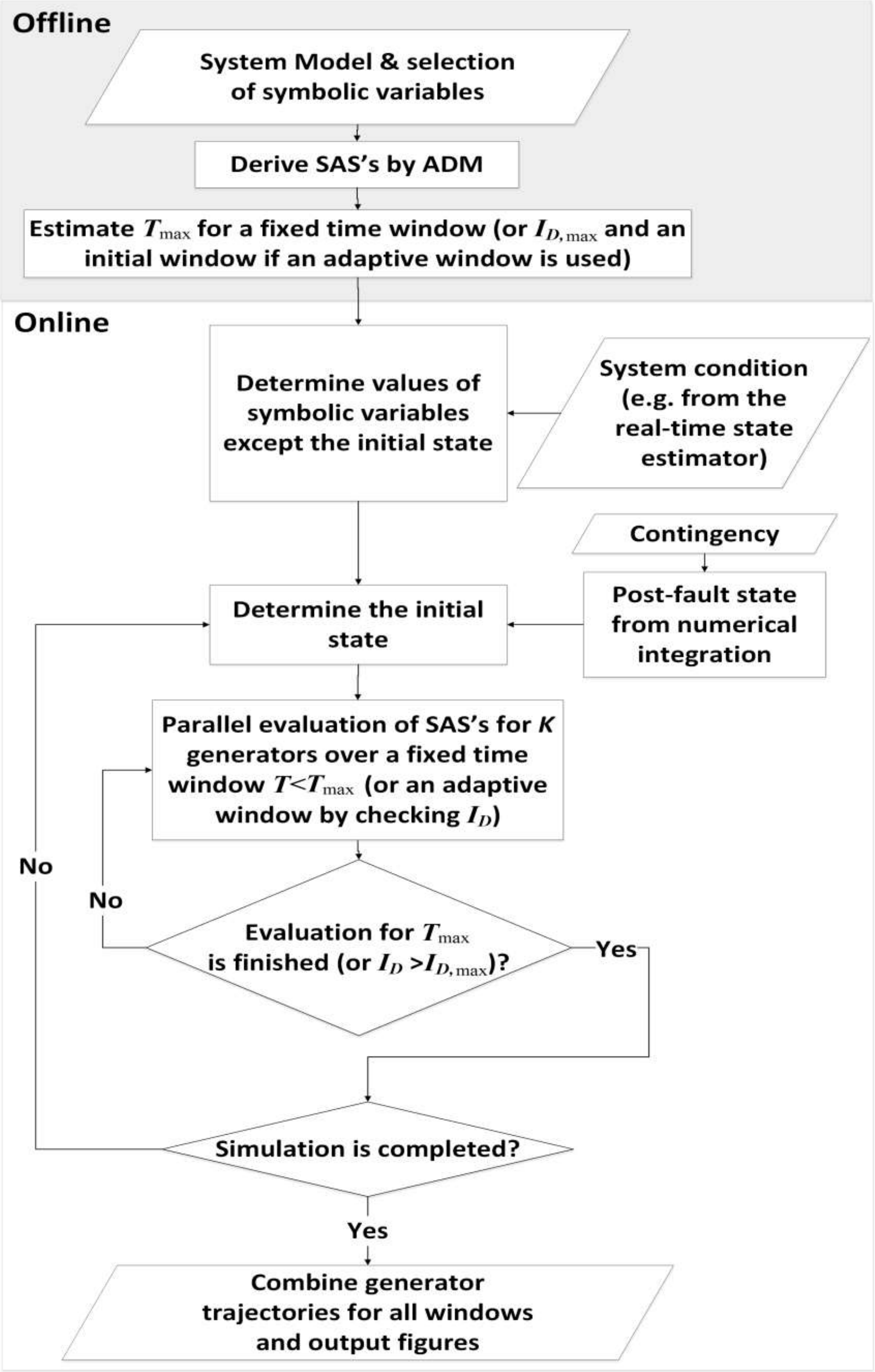

Offline
System Model & selection of symbolic variables
Derive SAS's by ADM
Estimate $T_{\max}$ for a fixed time window (or $I_{D,\max}$ and an initial window if an adaptive window is used)
Online
Determine values of symbolic variables except the initial state
System condition (e.g. from the real-time state estimator)
Contingency
Post-fault state from numerical integration
Determine the initial state
Parallel evaluation of SAS's for K generators over a fixed time window $T<T_{\max}$ (or an adaptive window by checking I_D)
No
No
Evaluation for $T_{\max}$ is finished (or $I_D > I_{D,\max}$)?
Yes
Simulation is completed?
Yes
Combine generator trajectories for all windows and output figures

Fig. 2.7. Flowchart of the proposed approach.

If an adaptive time window for SAS evaluation is used, the offline stage also needs to estimate $I_{D,\max}$. The detailed implementation of estimating $I_{D,\max}$ is illustrated in section 2.3. If a fixed window is adopted, T can be chosen less than the minimum $T_{\max}$ estimated by a procedure similar to that for the determination of $I_{D,\max}$ using a list of scenarios.

2.2.2 Online Stage

For a specific contingency scenario, this stage evaluates the corresponding SAS's of every generator consecutively over time windows T, fixed or adaptive, until making up the expected simulation period. The first time window needs to know the post-contingency initial system state, which can be obtained from numerical integration for the fault-on period until the fault is cleared. Starting from the second window, the initial state takes the final state of the previous window.

If an adaptive time window is applied, an initial window may be chosen less than the estimated $T_{\max}$ for a fixed window. Then, during each window, the divergence indicator I_D for each state variable is calculated and compared with the threshold $I_{D,\max}$ acquired in the offline stage in order to decide when to proceed to the next window, i.e. the end of the current window. Thus, even if the initial window is not small enough, comparison of I_D and $I_{D,\max}$ will enable self-adaptive adjustment of the window.

Within each window, because SAS's are independent expressions, their evaluations can be performed simultaneously on parallel computers. In expression, each SAS is the sum of terms in this form

$$C\cdot\underbrace{x_i\ldots x_j}_{h}t^n\underbrace{f_k(x_k)\ldots f_l(x_l)}_{m}\ \text{where}\, f(\cdot)\text{ is }\sin(\cdot)\text{ or }\cos(\cdot) \qquad (2\text{-}20)$$

Where C is a constant which depends on system parameters, t is time, i, j, k and l are integer indices of state variables. For different numbers of SAS terms and different systems, the ranges of h, m and n are different. For the IEEE 39-bus system with 3 SAS terms tested in Section 2.3, h=0,…,3, n=0,1,2 and m=0,…,4. Expression (2-20) is defined as one Computing Unit (CU) in this dissertation. All such CUs can be evaluated simultaneously on parallel processors to accelerate the online stage.

The proposed SAS-based approach may be applied for fast power system simulation in the real-time operating environment: in the offline stage, an SAS is derived that symbolizes a group of uncertain parameters like *Group-2*; then, in the online stage, whenever the real-time state estimation is finished (typically, every 1 to 3 minutes) to give the current power-flow solution and network topology, the SAS will be evaluated to provide simulation results on a given contingency. However, if a change on the network topology or any parameter about the operating condition is detected in real time by, e.g., the SCADA system [34] and makes the most recent state estimation result invalid, the SAS evaluation should wait until the state estimator gives a new estimation result. Thus, online power system simulation using the proposed approach can be performed synchronously with real-time state estimation.

2.3 Case Studies on the IEEE 39-bus System

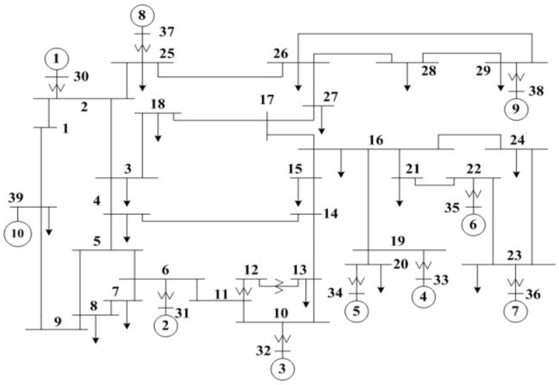


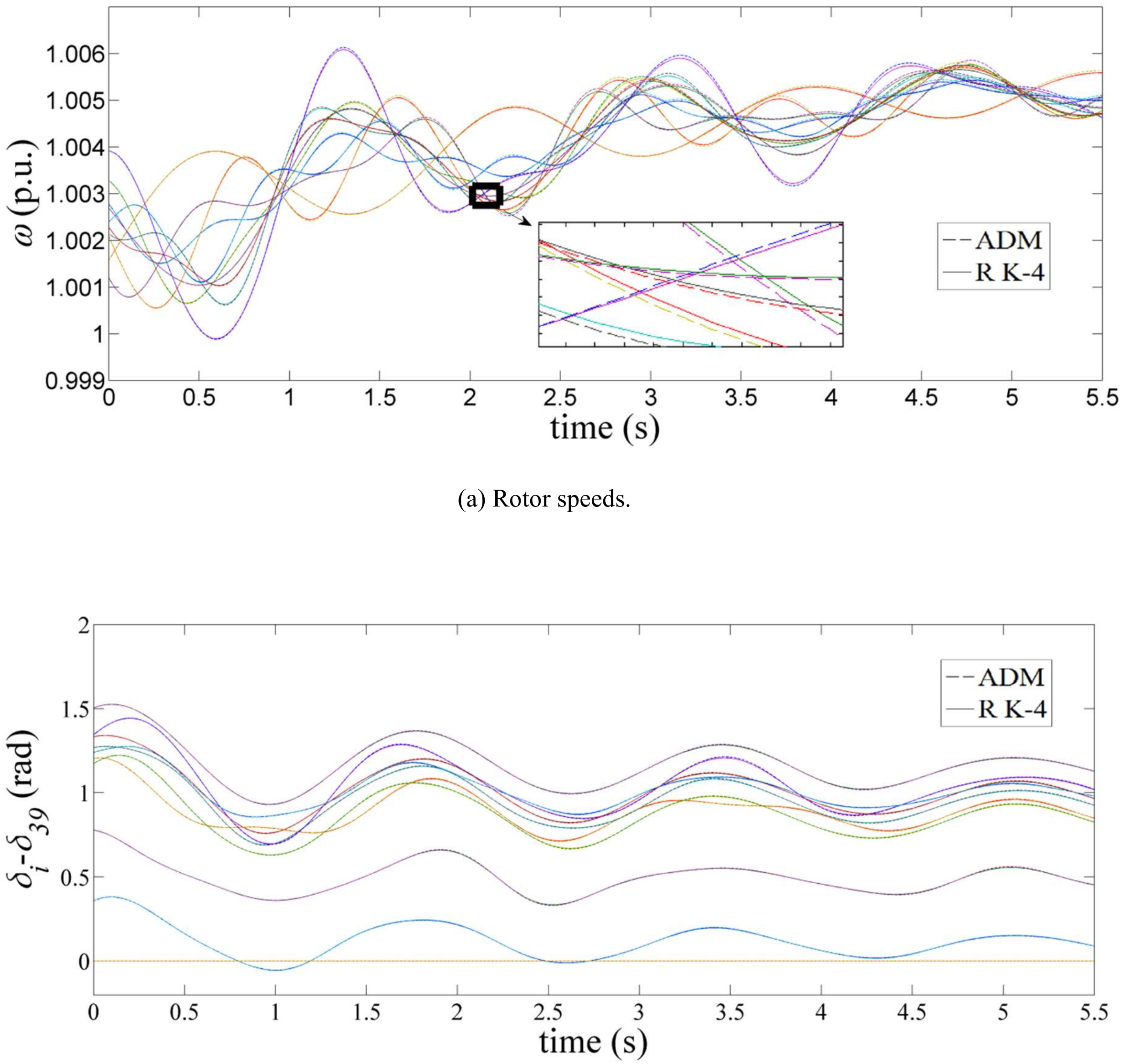
Fig. 2.8. IEEE 10-generator 39-bus system.

IEEE 10-generator, 39-bus system, as shown in Fig. 2.8, is used to validate the SAS-based approach for power system simulation. Generator 39 has the largest inertia and its rotor angle is defined as the reference. The proposed two-stage scheme is tested using both a fixed time window and an adaptive time window.

2.3.1 Fixed Time Window

A permanent three-phase fault lasting for 0.08 s is applied to line 3-4 at bus 3. We preset $I_{D,\max}$=0.005 p.u. (per unit) for all state variables except for the rotor angle. If all generators are represented by the 4th-order model in (2-9), our tests show that when an SAS with 2 terms is evaluated over a time window of 0.002 s, the largest 2nd SAS term of the state variables is 0.0047 p.u. $<I_{D,\max}$, which means $T_{\max}\approx$0.002 s for a 2-term SAS. Fig. 2.9 gives the results from the M-ADM (dash lines) using a 0.001 s window and the results from the R-K 4 (solid lines) with a 0.001 s integration step, which are identical.

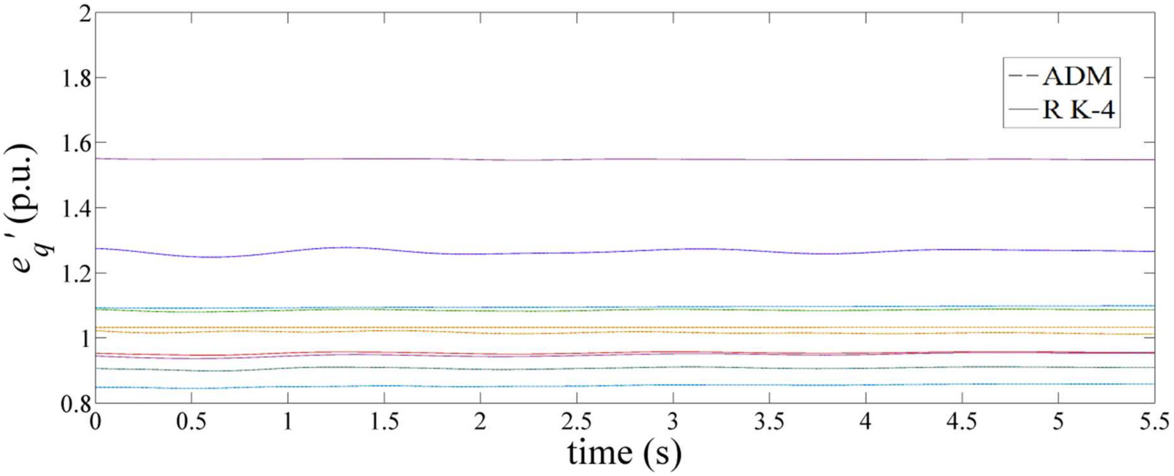
Fig. 2.9. Comparison of the simulation results given by the R-K 4 and the 2-term SAS using a fixed time window of 0.001 s.



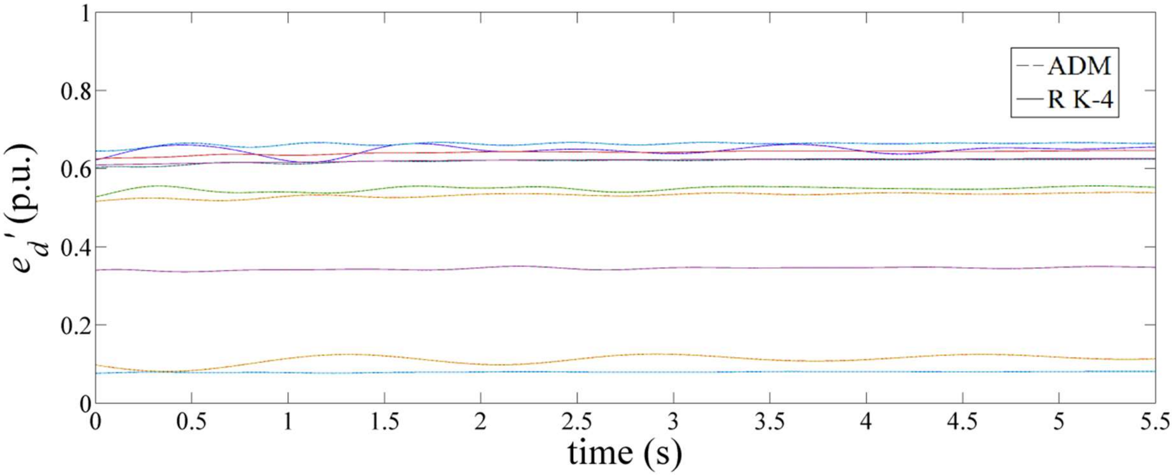
(a) Rotor speeds.

(b) Rotor angles.

Fig 2.9 continued



(c) q-axis transient voltages.



(d) d-axis transient voltages.

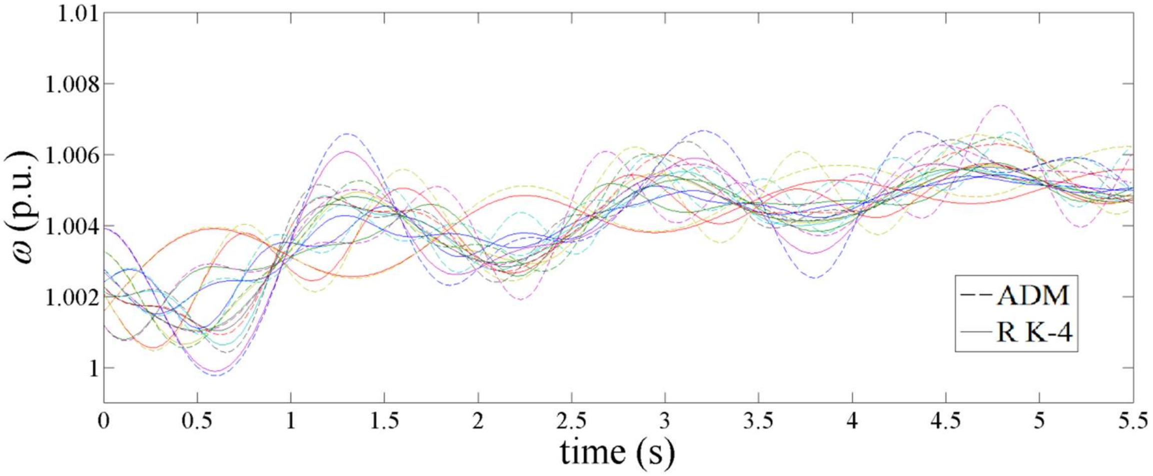
Fig 2.9 continued

If the time window and integration step are both increased to 0.01 s (>$T_{\max}$), the simulation results from the R-K 4 and M-ADM have slight, noticeable differences as shown in Fig. 2.10.

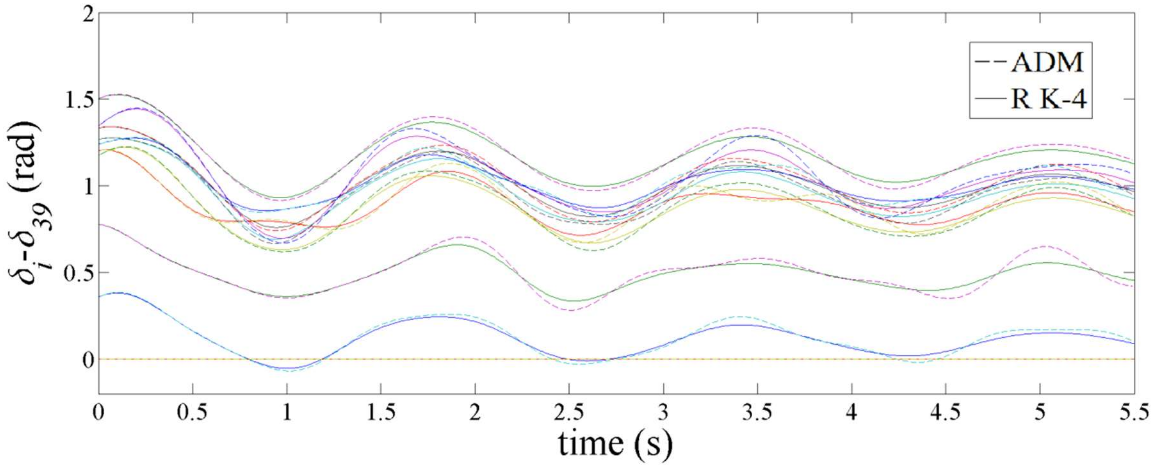
Although including more terms is expected to increase $T_{\max}$ as indicated by Fig. 2.1, using an SAS with 3 terms does not extend $T_{\max}$ significantly in this case. For example, use a 0.01 s time window to run a 3-term SAS for the same contingency, there are still obvious mismatches between the R-K 4 and M-ADM results.

Moreover, a 3-term SAS has a more complex expression, so it takes longer to evaluate than a 2-term SAS. When an SAS is evaluated over a fixed time window T for power system simulation, the last SAS terms, i.e. divergence indicator I_D's, of all state variables can distinguish numerical instability from power system instability: if the simulated system trajectory becomes unstable while all I_D's are still small, e.g. much less than the predefined $I_{D,\max}$, it is very likely to be power system instability; if some I_D also increases drastically to approach or exceed $I_{D,\max}$ when the system trajectory appears to be unstable, numerical instability may happen. Thus, a smaller T should be used to re-evaluate the SAS for verification of numerical instability.

Fig. 2.10. Comparison of the simulation results given by the R-K 4 and the 2-term SAS using a fixed time window of 0.01 s.

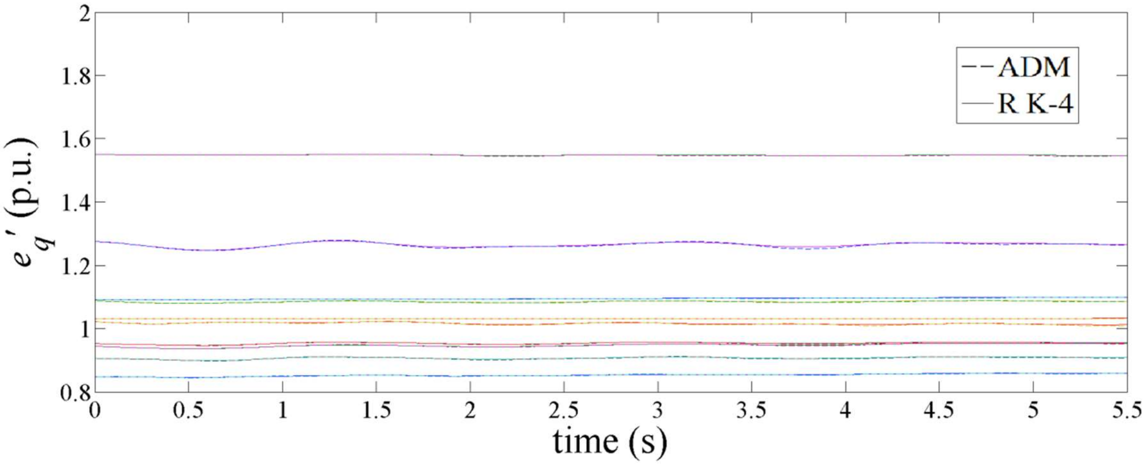


(a) Rotor speeds.

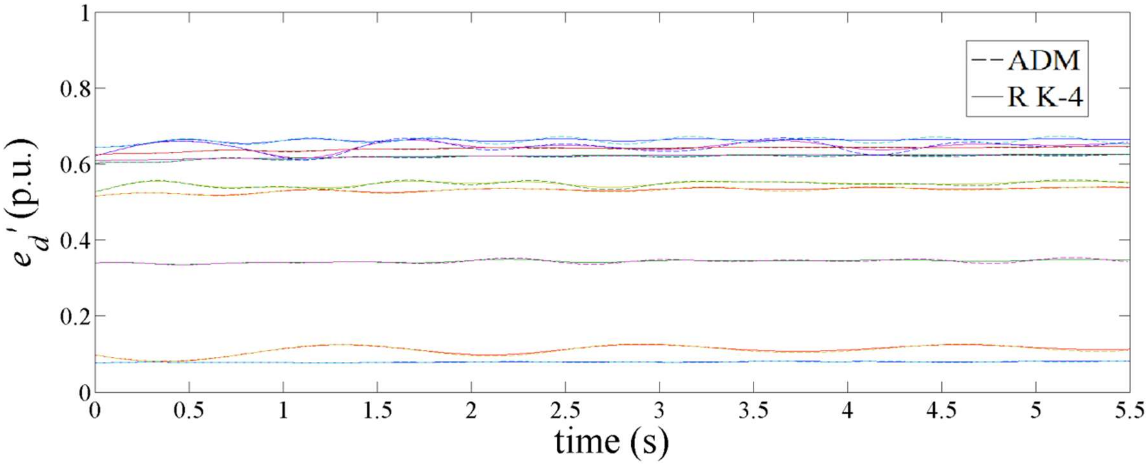


(b) Rotor angles.

Fig 2.10 continued



(c) q-axis transient voltages.



(d) d-axis transient voltages.

Fig 2.10 continued

For example, if T is increased to 0.02 s $\approx 10\times T_{\max}$, the simulation results diverge with numerical instability introduced on purpose as shown in Fig. 2.11, where the results from the R-K 4 method are still stable. That numerical instability can be detected by I_D's $>I_{D,\max}$ for many windows.

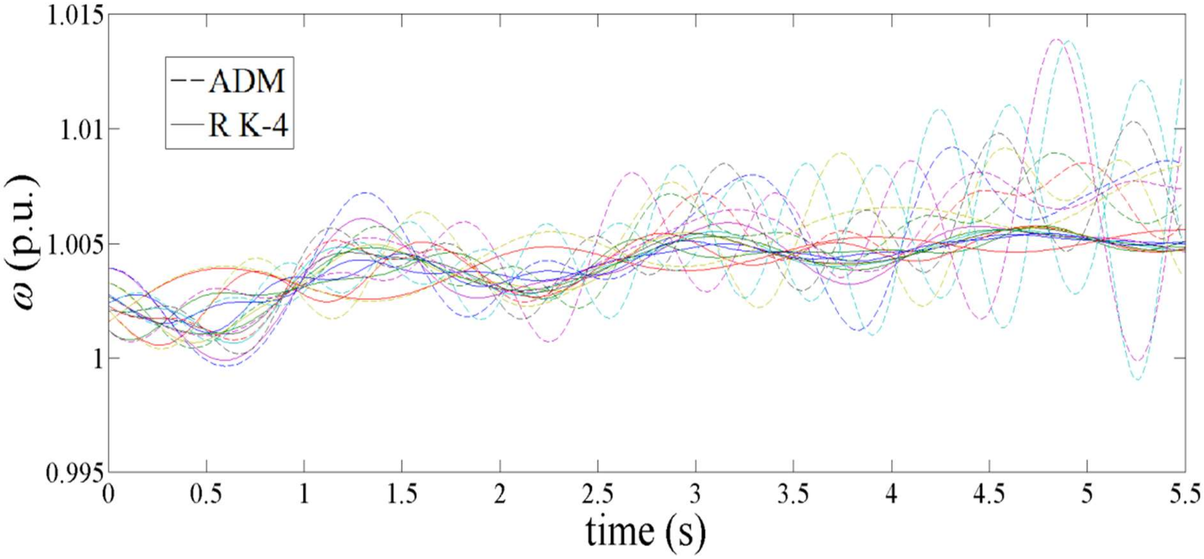


Fig. 2.11. Comparison of the simulation results of rotor speeds given by the R-K 4 and the 2-term SAS using a fixed time window of 0.02 s.

From the results of Fig.2.9 to Fig. 2.11, as T increases from 0.001 s to 0.01 s and then to 0.02 s, the largest I_D of all states variables increases from 0.0023 p.u. to 0.0279 p.u. (i.e. 12.1 times) and then to 0.1051 p.u. (i.e. 45.7 times), which indicates the occurrence of numerical instability. I_D can be utilized to avoid numerical instability by changing the time window adaptively. The detailed method will be proposed in the next sub-section.

The M-ADM is also tested on the system having each generator represented by the 6th-order DE model in (2-9), (2-11) and (2-12) containing the exciter and governor. The parameters of exciters and governors are set up as T_{Ak} = 0.02 s, K_{Ak} = 5, T_{gk} = 0.5 s, R_k = 0.01 for all machines. A 2-term SAS is derived for each of the six state variables, and the time window is selected to be 0.001 s within the estimated $T_{\max}$. Under the same contingency on line 3-4, The R-K 4 simulation indicates the frequency oscillation is better damped than that without a governor. Fig. 2.12 compares the results from the M-ADM (dash lines) and R-K 4 (solid lines) for each state variable, which match well.

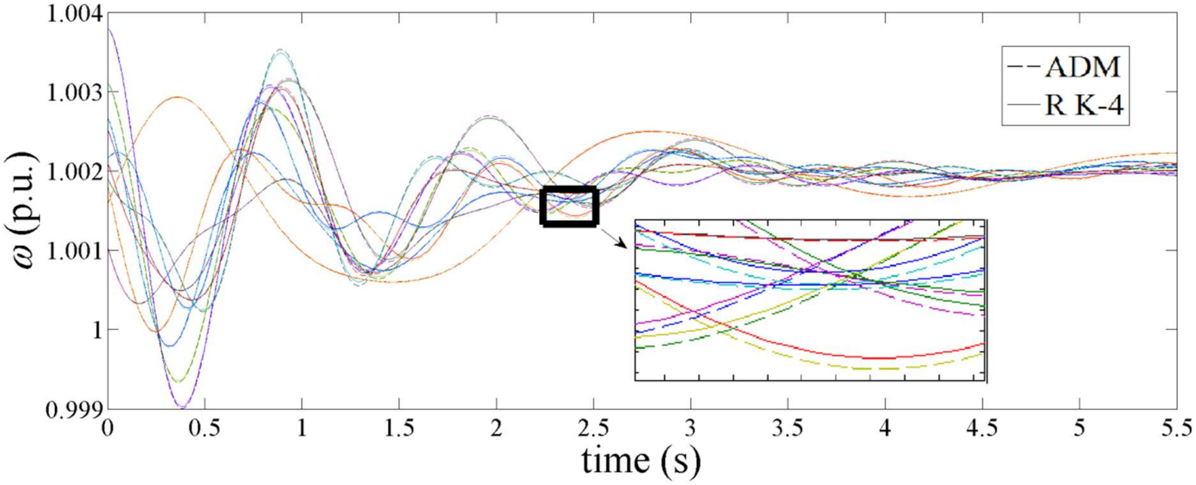
2.3.2 Adaptive Time Window

The first step is to use a list of contingencies to determine an $I_{D,\max}$ that can guarantee the accuracy of an SAS and avoid numerical instability in simulation by the M-ADM. For the illustration purpose, the above contingency on line 3-4 and a second contingency adding a three-phase fault lasting 0.08 s on line 15-16 at bus 15 are considered.

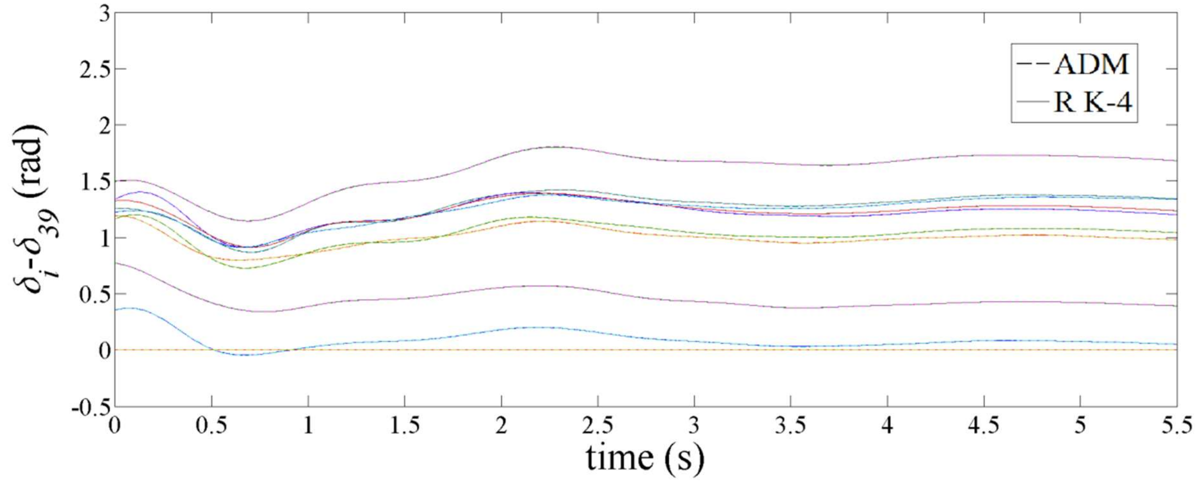
In practical studies, all the possible contingencies including line faults, bus faults and generator outages at different locations with different durations should be simulated to determine an $I_{D,\max.}$

Although simulating all the possible contingencies takes considerably long time, such exhaustive study only need to be performed once. After the value of $I_{D,\max}$ is obtained, it can be used online for simulating all the contingencies for this system.

Fig. 2.12. Comparison of the simulations using the 6th-order generator model by the R-K 4 and the 2-term SAS using a fixed time window of 0.001 s.

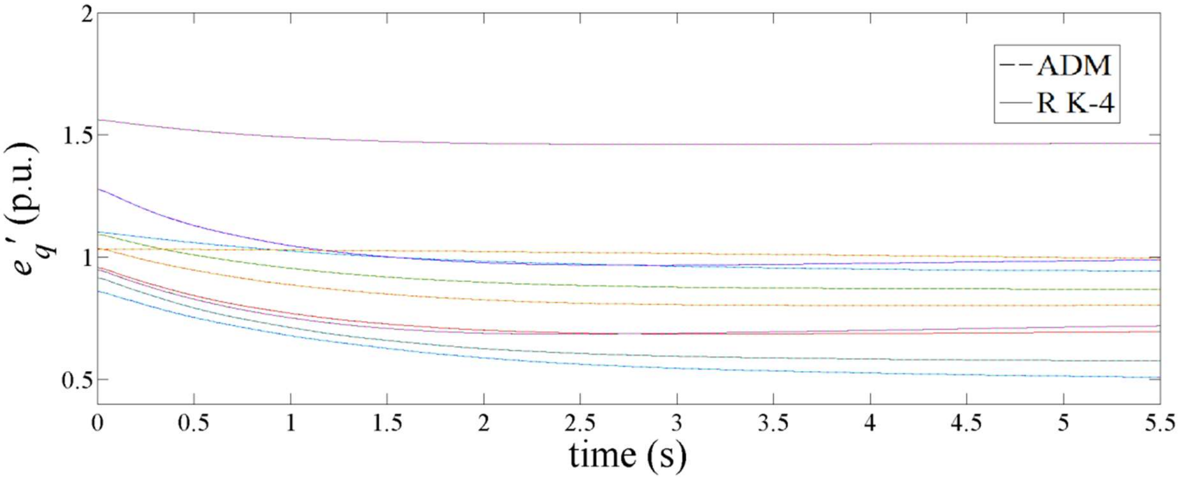


(a) Rotor speeds.

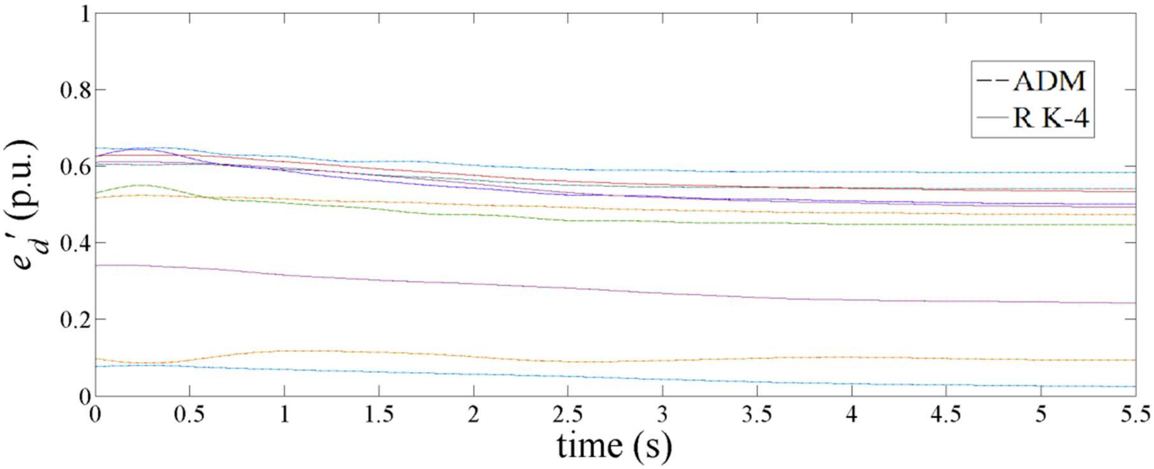


(b) Rotor angles.

Fig 2.12 continued

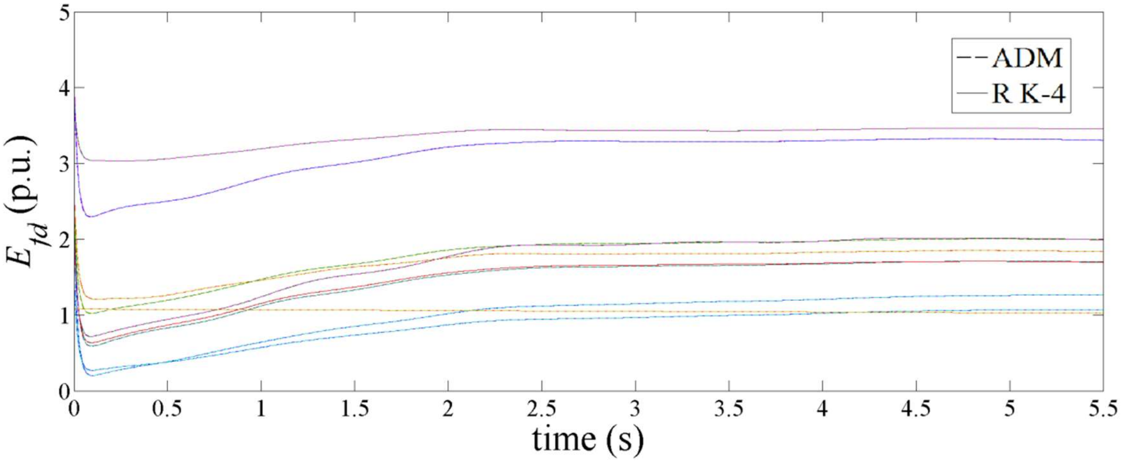


(c) q-axis transient voltages.

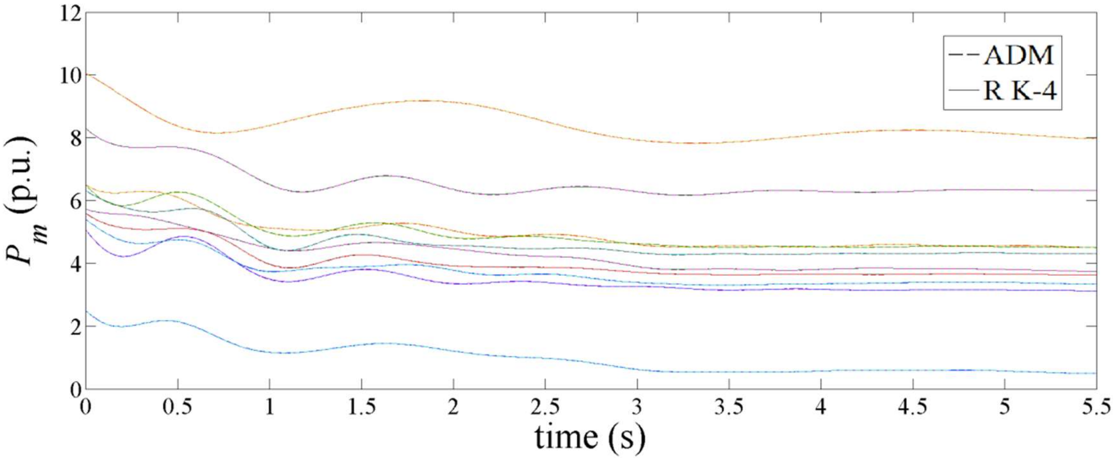


(d) d-axis transient voltages.

Fig 2.12 continued



(e) Field voltages.



(f) Governor outputs.

Fig 2.12 continued

Consider the 3rd SAS term of each state variable (except the rotor angle) in per unit as an I_D. Fig. 2.13 plots the I_D's for all those state variables of 10 generators, where 3 random variations are added and the resulting trajectories are also plotted in the same figure. The effective $I_{D,\max}$ for two contingencies are found both associated with $|e'_{d5,2}|$, which are 6.5×10^{-6} and 9.4×10^{-6} (p.u.), respectively.

Fig. 2.14 gives the result from a 3-term SAS evaluated over an adaptive time window, which is identical to the R-K 4 result.

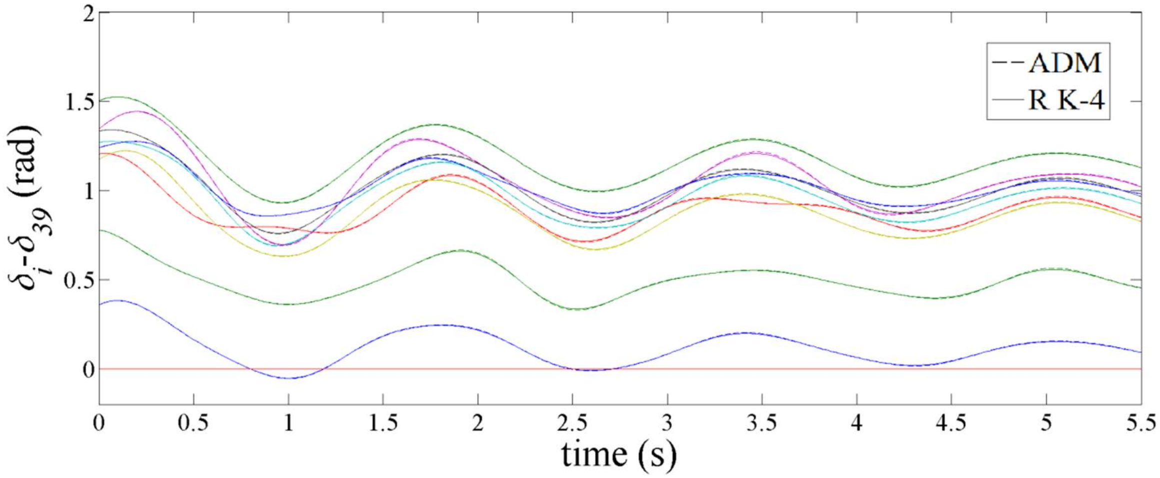
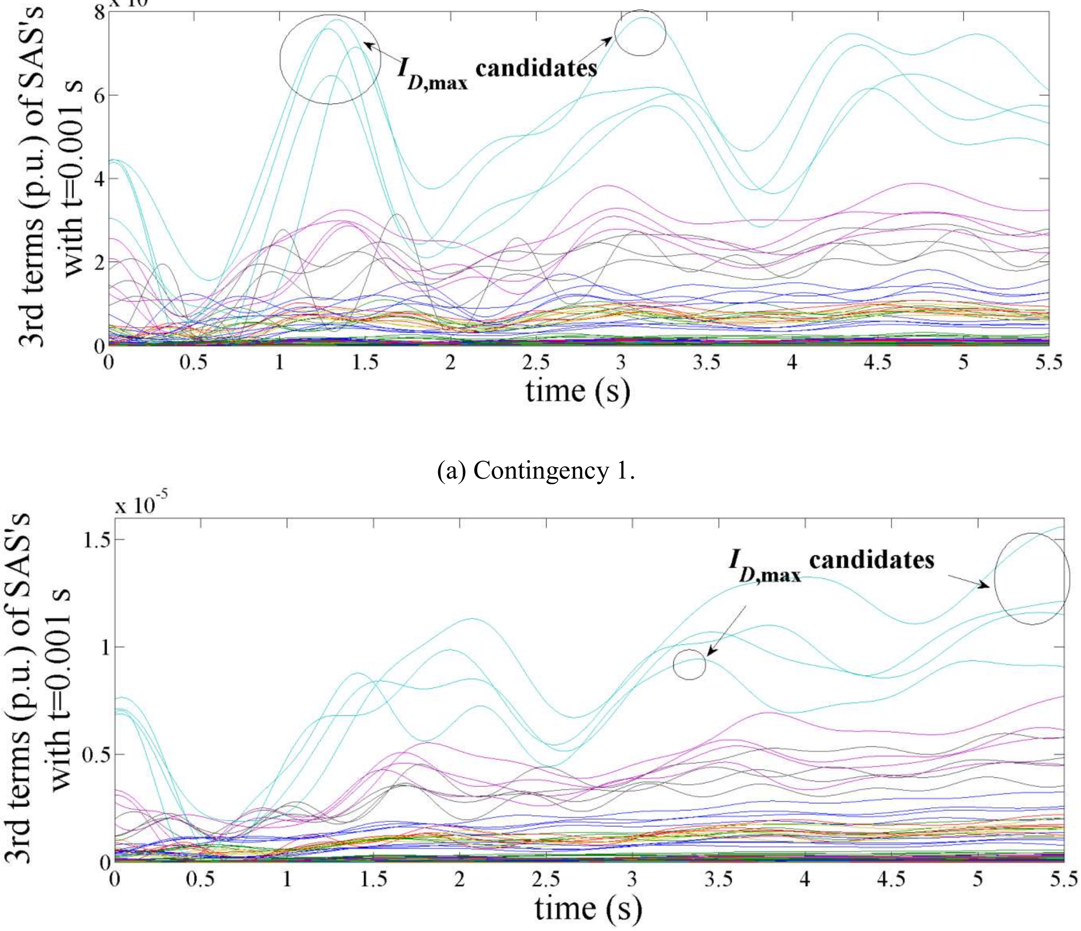


Fig. 2.13. Comparison of rotor angles given by the R-K 4 and the 3-term SAS using an adaptive time window initiated from 0.001 s.



(a) Contingency 1.

(b) Contingency 2.

Fig. 2.14. Estimation of $I_{D,\max}$.

Fig. 2.15 plots how the length of the time window changes with time during a 5.5-s simulation for three cases: 1) the 2-term SAS with an initial *T*=0.001 s, 2) the same SAS with an initial *T*=0.01s, and 3) the 3-term SAS with an initial *T*=0.001 s. The comparison of the cases 1) and 2) in Fig. 2.15 verifies that, if an adaptive time window is used, the accuracy of simulation is independent of the choice of the initial time window since the *T* of the case 2) adaptively decreases below 0.002 s soon after simulation starts. For the cases 1) and 2), the largest *T* reaches 0.0022 s.

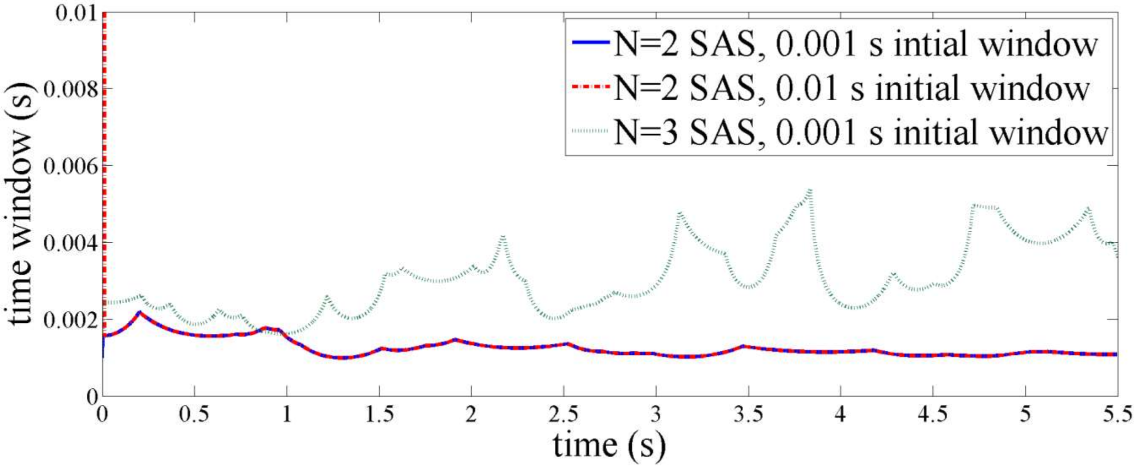


Fig. 2.15. Adaptive changing of time window length.

A main advantage of using an adaptive time window is that the total number of windows for evaluation is effectively reduced. The M-ADM using a fixed 0.001 s window evaluates 5500 windows to finish 5.5-s simulation while the case 1) using an adaptive time window only takes 4500 windows (i.e. 4500/5500=81.8%) to finish the same simulation period. For the case 3), the reduction of time windows is even more significant. As shown in Fig. 2.15, the largest T reaches 0.005 s, which is more than twice of the largest T for the 2-term SAS. Also, the total number of windows drops to 2000 (i.e. 2000/5500=36.4%). Thus, a conclusion is that using an adaptive time window enables the M-ADM to better exploit the advantage with a higher order SAS in terms of the reduction of the window number.

In the future development of a practical M-ADM based power system simulation tool, the optimal size of the time window and the proper number of SAS terms should be decided in a more adaptive way based on the information of the simulated power system to minimize the user intervention. It is not the focus of this dissertation but will be addressed in the future work.

2.3.3 Time Performance

To demonstrate the time performance of the proposed SAS-based approach, the following three cases are tested:

- ***Case-A**:* only symbolizing time t and initial state variables, i.e. for one specific simulation.
- ***Case-B**:* beside *Case-A*, also symbolizing the reduced admittance matrix **Y** about 10 generator EMFs, i.e. for simulating different faults under one specific loading condition. Magnitudes and angles of elements of the reduced admittance matrix are symbolized separately to generate two symmetric symbolic 10×10 matrices.
- ***Case-C**:* beside *Case-B*, also symbolizing generators' mechanical powers to make the SAS be also good for simulating various loading conditions.

Here, the load at each bus is represented by a constant impedance load model and is embedded in the reduced admittance matrix **Y**. In the online stage, for a given power-flow condition with all loads known, load impedances will first be calculated, and then with the knowledge of the post-fault network topology, all elements of **Y** can be calculated in order to evaluate the SAS.

The offline stage is implemented in MAPLE and the online stage is performed in MATLAB. For 4th-order and 6th-order generator models, the numbers of CU's comprising the 3-term SAS's of each state variable are given in Tables 2.3 and 2.4, respectively, for three cases.

Table 2.3. The Number of CUs for the 4th-Order Model System.

State Variable	Case-A	Case-B	Case-C
ω_k	4,269	11,430	11,430
δ_k	150	150	150
e'_{qk}	225	301	301
e'_{dk}	223	299	299

Table 2.4. The Number of CUs for the 6th-Order Model System.

State Variable	Case-A	Case-B	Case-C
ω_k	4,272	11,434	11,434
δ_k	150	150	150
e'_{qk}	227	303	303
e'_{dk}	223	299	299
E_{fd}	2,644	5,234	5,234
P_m	153	155	155

For *Case-A*, it only takes less than 3 μs to evaluate one CU. If all such CU's are evaluated simultaneously on parallel processors, it takes about 3 μs to evaluate one SAS for each time window plus the time costs for communication in parallel computing. Because summating the values of all CU's for a state variable is essentially the addition of constants, it is extremely fast. The additions for different state variables can also be performed in parallel. Thus, the final time for summating all CUs equals the time for the most complex SAS expression, often on a rotor speed, which only takes 7 μs. Therefore, the ideal total time cost for evaluations of state variables of one generator is 3+7=10 μs per time window *T*. If evaluations for various generators are also done simultaneously on an unlimited number of parallel processors, that time is also the time cost τ for SAS evaluation over each time window *T*. The R-K 4 method takes 0.37 s to finish a 5.5-s simulation with all generators represented by the 4th-order model on one computer processor. (It takes 0.48 s if all generators are represented by the 6th-order model.) Given the fact that a 3-term SAS only needs 2000 adaptive time windows for a 5.5-s simulation, it can be concluded that the online stage ideally only takes 0.00001×2000=0.02 s to finish simulation on parallel processors, which is about 18 times faster than the time cost of the R-K 4. Ratio T/τ=5.5/0.02=275, i.e. the number of times faster than wall-clock time. For *Case-B* and *Case-C*, T/τ=137.5 as given by Table 2.4, which indicates how many times the simulation can be faster than the wall-clock time.

By comparing Tables 2.3 and 2.4, it can be easily noticed that even after the exciter and governor models are added, the state variables that have the most CUs are still rotor speeds. Meanwhile, the number of CUs of each rotor speed's SAS only increases very slightly (by 3 for Case-A and 4 for Case-B and Case-C.) when the generator model changes from the 4th-order to the 6th-order. Basically, adding those details or controllers to each generator does not influence the online performance of the proposed approach.

The time performance of the offline stage is not as critical as the online stage, so it is evaluated in a sequential computing manner. Tables 2.5 and 2.6 summarize the time performances of both offline and online stages for two systems respectively using the 4th and 6th order generator models under the assumption of an ideal parallel computing capability.

Table 2.5. Time Performance on the 4th-Order model System.

	Case-A	*Case-B*	*Case-C*
Offline time cost (s)	198.05	682.18	711.17
Online time cost (s)	0.02	0.04	0.04
Ratio T/τ	275.0	137.5	137.5

Table 2.6. Time Performance on the 6th-Order model System.

	Case-A	*Case-B*	*Case-C*
Offline time cost (s)	6215.51	13472.91	16339.71
Online time cost (s)	0.02	0.04	0.04
Ratio T/τ	275.0	137.5	137.5

Considering that the number of parallel processors cannot be infinity in practice, we also studied how the time performance changes with the number of available processors. As theoretical estimates, ideal parallelism among all available processors is assumed. Thus, all processors are assumed to take equal computational burdens. The results are listed in Table 2.7 for *Case-A* using 3-term SASs. From the table, when the number of processors drops to 100, the simulation time increases to 0.3 s, which is close to 0.37 s of the R-K 4. If the number of parallel processors is further decreased, the simulation using the M-ADM becomes slower than the R-K 4.

Table 2.7. Influence of Parallel Capability on Time Performance.

Number of Parallel Processors	Time Cost of Each Time Window (s)	Time Cost for a 5.5-s simulation (s)
∞	1.0×10^{-5}	2.0×10^{-2}
1000	1.4×10^{-5}	2.8×10^{-2}
100	1.5×10^{-4}	3.0×10^{-1}
10	1.5×10^{-3}	3.0

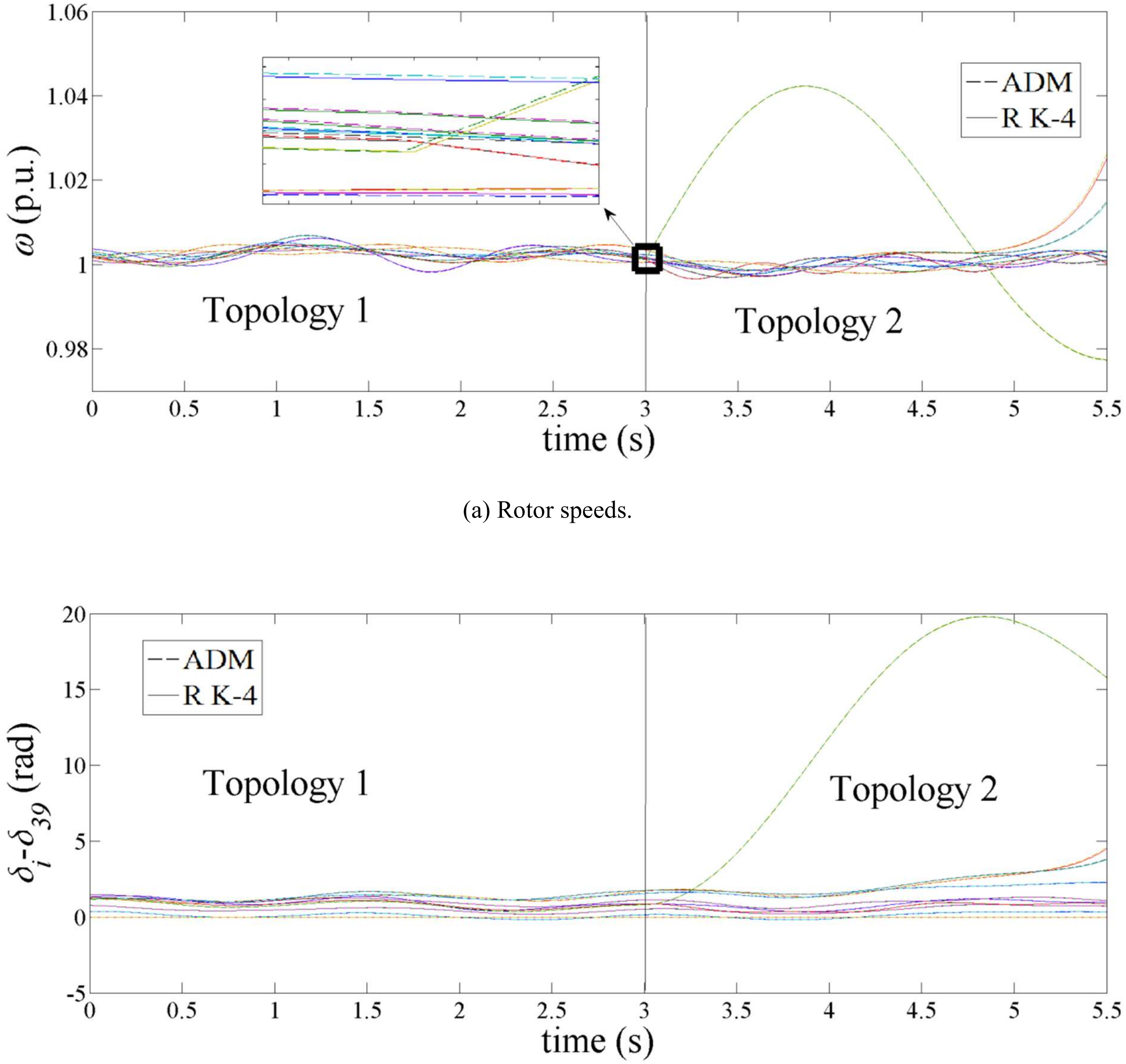
When a long list of contingencies need to be simulated, parallel processors may simulate multiple contingencies simultaneously, so power system simulation using the proposed SAS-based approach will be parallelized also at the contingency level besides the aforementioned CU level. Thus, a more sophisticated hierarchy for parallel implementation of the proposed SAS-based approach should be designed and will be addressed in the future work.

2.3.4 Simulation of a contingency with multiple disturbances

The proposed SAS-based approach can be used to simulate a contingency containing multiple disturbances, e.g. "n-1-1" and even "n-k" contingencies, which involve one or more disturbances during the simulation period. The same SAS can be used for the entire simulation period as long as all parameters that may change during the simulation period are defined as symbolic variables like an SAS from *Case-B* or *Case-C*.

In the following, we demonstrate how to use the SASs of *Case-B* to perform an "*n*-1-1" simulation involving a topological change of the system during the simulation period. The 6th-order generator models are adopted. The initial contingency is still the same as that in Fig. 2.9-Fig. 2.12 except that at *t*=3 s, the line 22-35 is opened, making the system have a different topology in the remaining 2.5 s. The SAS's derived for *Case-B* treat all elements of reduced **Y** matrix as symbolic variables. Therefore, at *t*=3 s, the time when topology changes, new values of the elements in the reduced **Y** matrix should be plugged into the SASs. The simulation results are shown in Fig. 2.16. Generator 35 loses its stability. The online time cost is 0.04 s with ideal parallelism on sufficient processors.

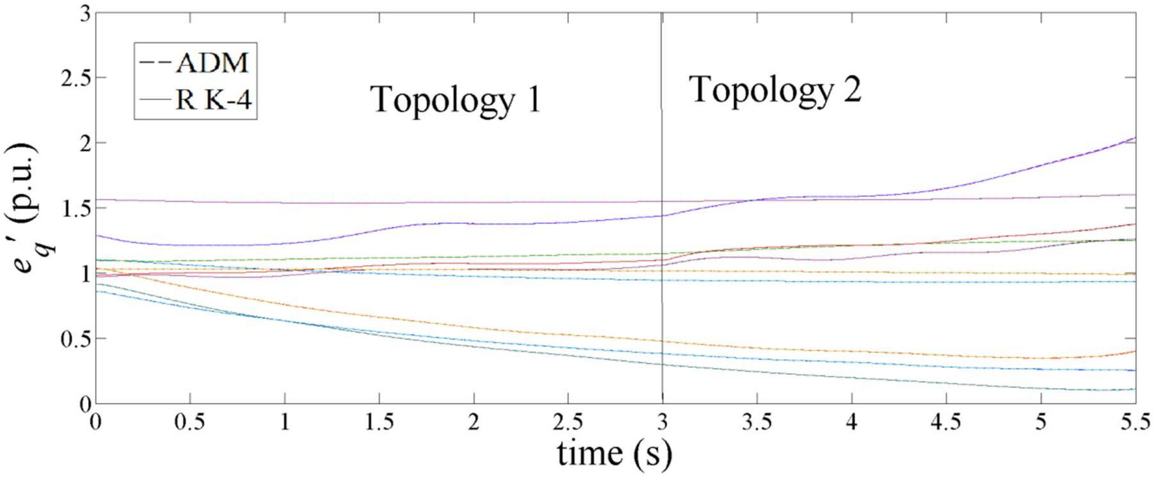
Fig. 2.16. Comparison of the simulation results with a topology change at t=3 s given by the R-K 4 and a 3-term SAS using an adaptive time window.



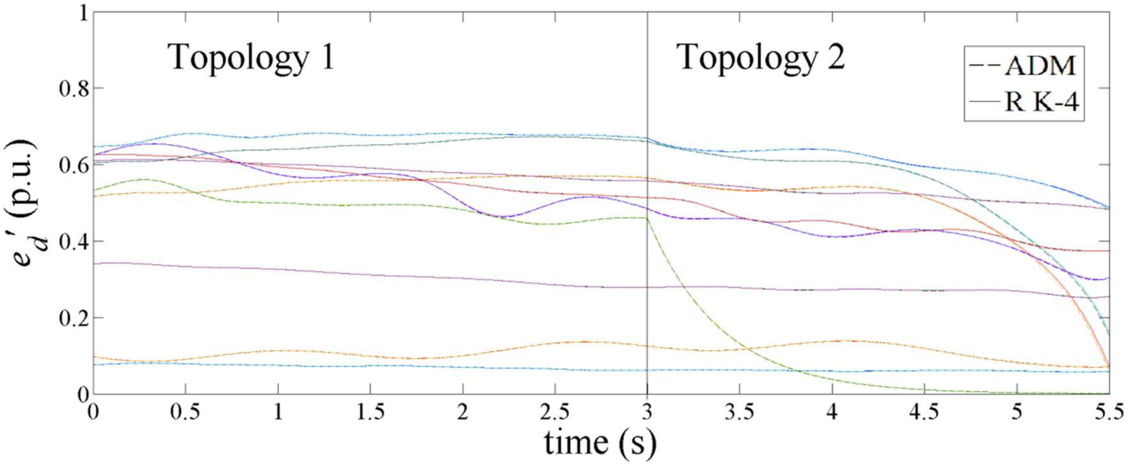
(a) Rotor speeds.

(b) Rotor angles.

Fig. 2.16 continued

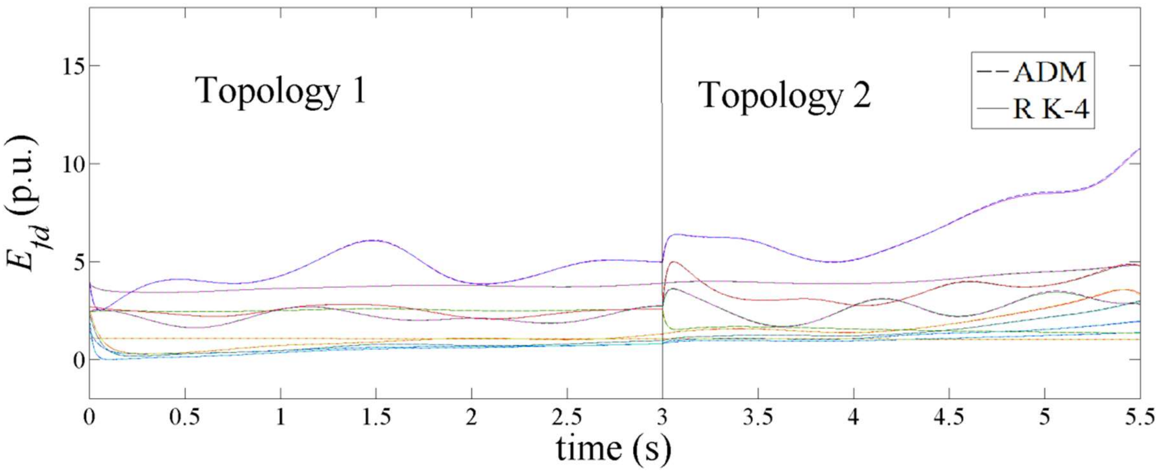


(c) q-axis transient voltages.

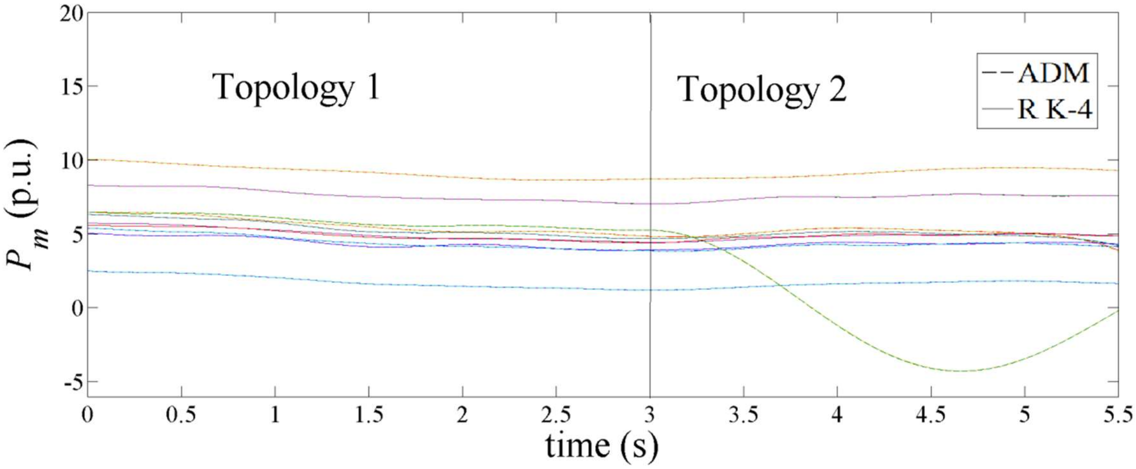


(d) d-axis transient voltages.

Fig. 2.16 continued



(e) Field voltages.



(f) Governor outputs.

Fig. 2.16 continued

2.3.5 Simulating Systems with Non-conforming Load

The proposed SAS-based approach also can be applied to simulating systems which have ZIP loads. The challenge of considering ZIP loads in power system simulation is that the system DAEs can not be simplified as differential equations anymore. The ZIP load characteristic is dependent on the voltage at its terminal bus as shown in (2-21) and (2-22).

$$P_i = P_{0,i}\left(p_1 \left(\frac{\left|V_{nc,i}\right|}{V_{nc0,i}} \right)^2 + p_2 \left(\frac{\left|V_{nc,i}\right|}{V_{nc0,i}} \right) + p_3 \right) \tag{2-21}$$

$$Q_i = Q_{0,i}\left(q_1 \left(\frac{\left|V_{nc,i}\right|}{V_{nc0,i}} \right)^2 + q_2 \left(\frac{\left|V_{nc,i}\right|}{V_{nc0,i}} \right) + q_3 \right) \tag{2-22}$$

Therefore the buses which connected to ZIP can not be eliminated while calculating reduced admittance matrix.

$$\begin{bmatrix} \mathbf{I}_{nc} \\ \mathbf{I}_g \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_{11} & \mathbf{Y}_{12} & \mathbf{Y}_{13} \\ \mathbf{Y}_{12}^{\mathrm{T}} & \mathbf{Y}_{22} & \mathbf{Y}_{23} \\ \mathbf{Y}_{13}^{\mathrm{T}} & \mathbf{Y}_{23}^{\mathrm{T}} & \mathbf{Y}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{V}_{nc} \\ \mathbf{\Psi} \\ \mathbf{V}_{bus} \end{bmatrix} \tag{2-23}$$

Voltages of the buses with no load or constant impedance load can be eliminated using (2-24),

$$\mathbf{V}_{bus} = -\mathbf{Y}_{33}^{-1}\mathbf{Y}_{13}^{\mathrm{T}}\mathbf{V}_{nc} - \mathbf{Y}_{33}^{-1}\mathbf{Y}_{23}^{\mathrm{T}}\mathbf{\Psi} \tag{2-24}$$

Then the current injections into non-conforming load buses $\mathbf{I}_{\mathrm{nc}}$ and from generator terminal buses $\mathbf{I}_{\mathrm{G}}$ are,

$$\mathbf{I}_{nc} = \mathbf{Y}_{nc}\mathbf{V}_{nc} + \mathbf{Y}_{ncg}\mathbf{\Psi} \tag{2-25}$$

$$\mathbf{I}_{g} = \mathbf{Y}_{gnc}\mathbf{V}_{nc} + \mathbf{Y}_{g}\mathbf{\Psi} \tag{2-26}$$

where,

$$\mathbf{Y}_{nc} \triangleq \mathbf{Y}_{11} - \mathbf{Y}_{13}\mathbf{Y}_{33}^{-1}\mathbf{Y}_{13}^{\mathrm{T}}$$

$$\mathbf{Y}_{ncg} \triangleq \mathbf{Y}_{12} - \mathbf{Y}_{13}\mathbf{Y}_{33}^{-1}\mathbf{Y}_{23}^{\mathrm{T}}$$

$$\mathbf{Y}_{g} \triangleq \mathbf{Y}_{22} - \mathbf{Y}_{23}\mathbf{Y}_{33}^{-1}\mathbf{Y}_{23}^{\mathrm{T}}$$

$$\mathbf{Y}_{gnc} \triangleq \mathbf{Y}_{12}^{\mathrm{T}} - \mathbf{Y}_{23}\mathbf{Y}_{33}^{-1}\mathbf{Y}_{13}^{\mathrm{T}}$$

Moreover, the voltage of ZIP load terminal bus has to be solved numerically through Newton-Raphson method, since the analytical solution of algebraic equation (2-27) does not exist.

$$\mathbf{Y}_{ncg}\mathbf{\Psi} + \mathbf{Y}_{nc}\mathbf{V}_{nc} = \mathbf{I}_{nc} = \mathbf{I}_{cc} + \mathbf{I}_{cp} \tag{2-27}$$

where, the i-th element of $\mathbf{I}_{cc}$+$\mathbf{I}_{cp}$ is

$$I_{cc,i} + I_{cp,i} = -\left(\frac{p_3 P_{0,i} + p_2 P_{0,i}\dfrac{\left|V_{nc,i}\right|}{\left|V_{nc0,i}\right|} + j\left(q_3 Q_{0,i} + q_2 Q_{0,i}\dfrac{\left|V_{nc,i}\right|}{\left|V_{nc0,i}\right|}\right)}{V_{nc,i}} \right)^{*} \tag{2-28}$$

To consider non-conforming load, the algebraic equation about the current injection at generator terminal buses in (2-10) needs to be replaced by (2-26), which also considers the component of generator terminal bus currents due to the voltage of non-conforming load buses, i.e. $\mathbf{Y}_{gnc}\mathbf{V}_{nc}$.

Since the voltages at non-conforming load buses can not be solved analytically, they can be treated as symbolic variables in the derivation of an SAS at the offline stage and updated at the end of every window using the solution of (2-27) at the online stage. In a power system, there are generally more loads than generators, so instead of directly using (2-26) in the derivation of the SAS, which will introduce dim($\mathbf{V}_{nc}$) new symbolic variables, this implementation uses (2-29) to derive the SAS, which will introduce dim($\mathbf{I}_g$) < dim($\mathbf{V}_{nc}$) new symbolic variables.

$$\mathbf{I}_g = \mathbf{I}_{gnc} + \mathbf{Y}_g \mathbf{\Psi} \tag{2-29}$$

where $\mathbf{I}_{gnc}$ is the component of generator bus currents due to the voltages of non-conforming loads.

The procedure of simulating a power system with non-conforming load is outlined as follows:

1. Calculate the current injection from the non-conforming load to the generator buses $\mathbf{I}_{gnc}$.
2. Update the value of $\mathbf{I}_{gnc}$ in the SAS.
3. Evaluate the SAS for the upcoming window.

Using the 3rd order SAS and this procedure to simulate a 4-cycle 3-phase fault at bus 1 of the 10-machine 39-bus system with all loads represented by 20% constant impedance, 30% constant current and 50% constant power load, the simulation result of the rotor speed deviation is shown in Fig. 2.17 and the window length is shown in Fig. 2.18.

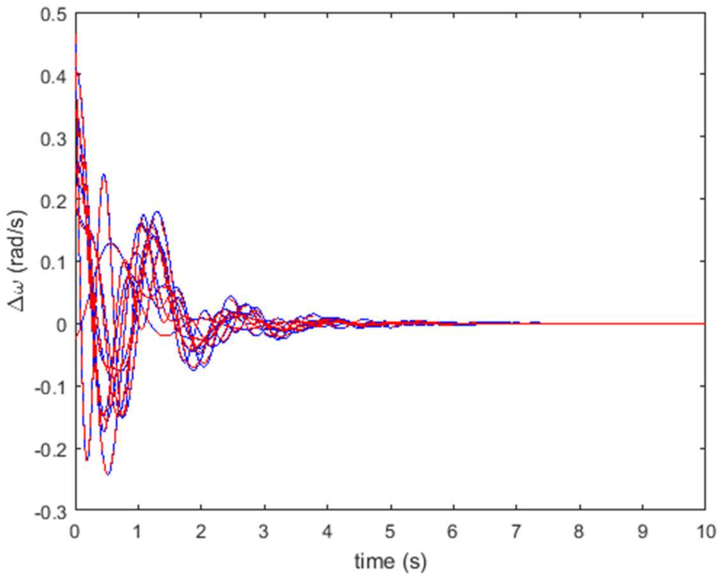


Fig. 2.17. The IEEE 10-machine 39-bus system with non-conforming load rotor speed deviation simulation comparison between the R-K 4 (blue) and SAS based approach (red).

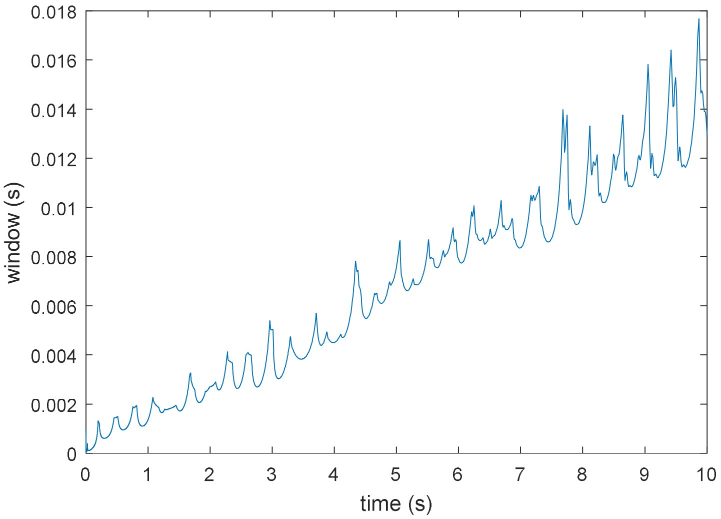


Fig. 2.18. The changing of time window length for the IEEE 10-machine 39-bus system with non-conforming load simulation.

CHAPTER THREE

SAS BASED STOCHASTIC POWER SYSTEM SIMULATION

Uncertainties exist in operations of power grids [35]. Many factors such as random load consumptions and unanticipated relay protection actions contribute to the randomness of grid operations. It can be foreseen that a future power grid will have more uncertainties and stochastic behaviors in system operations due to the increasing penetrations of responsive loads and intermittent renewable generations. Thus, dynamic security assessment (DSA) of power systems should be conducted in both deterministic and stochastic manners. However, most of today’s power system simulation software tools are still based on solvers of deterministic DAEs that do not involve stochastic variables to model uncertainties in system operating conditions.

3.1 Stochastic Simulation Approaches

In literature, there are three major approaches for the modeling of a dynamic system having stochastic effects as shown in Fig. 3.1: the master equation, the Fokker-Planck equation [36][37] and Gillespie method [38][39]. The master equation and the Fokker-Plank equation are widely applied in the field of computational biology, which both focus on the evolution of probability distribution; the Gillespie method focuses on individual stochastic trajectories. The first two approaches provide a more comprehensive understanding of stochastic effects with a dynamic system but require solving high dimensional partial differential equations, so they are computationally difficult to be applied to simulations of realistic power systems [40]. There have been works using the Gillespie method for power system simulation [41]-[44].

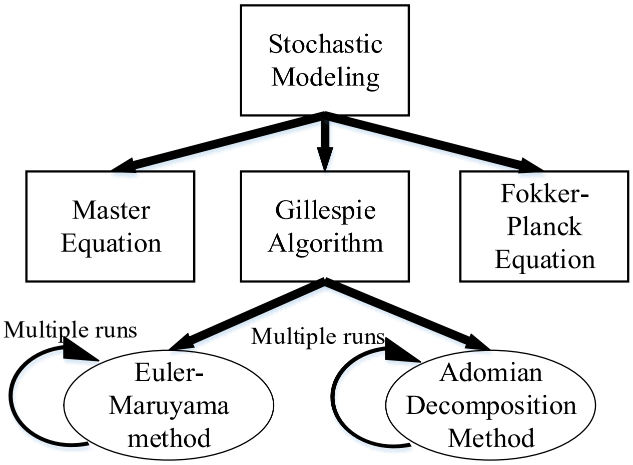


Fig. 3.1. Stochastic modeling approaches.

In recent years, some researchers have contributed to power system simulation in a less-deterministic manner. Reference [44] proposed a systematic method to simulate the system behaviors under the influence of stochastic perturbations on loads, bus voltages and rotor speeds. This approach introduces stochastic differential equations (SDEs) to represent stochastic perturbations and solves the equations by Ito calculus, and then a mean trajectory with the envelope on trajectory variations is yielded by repeating simulations for many times. Papers [42]-[44] utilize a similar approach to study power system stability under random effects. To analyze long term stability of a power system with wind generation, a new SDE model is developed in [45], which also applies the singular perturbation theory to investigate the slow dynamics of the system with stochastic wind generation. However, the time performance of such an approach based on Euler-Maruyama method can hardly meet the requirements for online power system simulation. Especially, when the penetration of distributed energy resources (DERs) reaches a high level, the distribution network behaves in a more stochastic manner as seen from the transmission network, and hence a large number of SDEs need to be included in the power system model, which will significantly influence the simulation speed. Also, the nature of the Gillespie method requires a large number of simulations on the same model to yield the mean trajectory as well as the envelope on variations. Therefore, adding any extra SDE to the existing set of SDEs will result in multiplying computing time by a factor of hundreds or even thousands.

In our previous work [28][46], a new semi-analytical approach for power system simulation has been proposed. That approach applies the Adomain decomposition method (ADM) to power system DAEs to derive a semi-analytical solution (SAS) for each state variable as an explicit function of symbolic variables including time, the initial system state and other selected parameters on the system condition; then each function is evaluated by plugging in values of its symbolic variables over consecutive small time windows to make up a desired simulation period so as to obtain the simulated trajectory of each state variable. Since the form of every SAS is a summation of finite terms for approximation, its evaluation can be fast and parallelized among terms. Thus, compared to traditional numerical integration based power system simulation, this semi-analytical approach decomposes the computation into offline derivation and online evaluation of an SAS and is better fit for online power system simulation and a parallel computing environment [46]. In fact, such a semi-analytical approach also suggests a viable, alternative paradigm for fast stochastic simulation. For example, early works by Adomian in the 1970s utilized the ADM to solve nonlinear SDEs [47] by embedding explicitly stochastic processes into the terms of an SAS.

For power system simulation in a stochastic manner, this dissertation proposes an approach as an extension of the ADM based approach proposed in [46]. Utilizing the semi-analytical nature of an SAS yielded by the ADM, this new approach embeds a stochastic model, e.g. a stochastic load model, into the SAS. Evaluation of an SAS with the stochastic model whose parameters are represented symbolically will not increase many computational burdens compared to evaluation of an SAS for deterministic simulation. Thus, an expected number of simulation runs for one single case are achieved by evaluating one SAS for the same number of times.

3.2 Power System SDE Model with Stochastic Loads

3.2.1 Synchronous Generator Modeling

For a power system having *K* synchronous generators, consider the 4th-order two-axis model (2-9) to model each generator having saliency ignored [22]. All generators are coupled through nonlinear algebraic equations (2-10) about the network.

3.2.2 Stochastic Load Modeling

A stochastic model can be built based on analysis on real data and assumptions on probabilistic characteristics of the stochastic variables. Traditionally, uncertainties in loads of a power system are ignored in time-domain simulation for the sake of simplicity.

However their stochastic behaviors are well-recognized in [48]. Taking stochastic loads into consideration will enable more realistic power system stability assessment.

This dissertation uses the Ornstein-Uhlenbeck process in [49] to model the stochastic variations of a load in these SDEs:

$$\dot{\mathbf{y}}_{\mathrm{PL}} = -\mathbf{a}_{\mathrm{P}} \circ \mathbf{y}_{\mathrm{PL}} + \mathbf{b}_{\mathrm{P}} \circ \mathbf{W}(t) \tag{3-1}$$

$$\dot{\mathbf{y}}_{\mathrm{QL}} = -\mathbf{a}_{Q} \circ \mathbf{y}_{\mathrm{QL}} + \mathbf{b}_{Q} \circ \mathbf{W}(t) \tag{3-2}$$

where **W**(t) is the white noise vector whose dimension equals the number of load buses, **a** and **b** parameters are drifting and diffusion parameters of the SDEs, operator "$\circ$" is the Hadamard Product, i.e., element-wise multiplication, and $\mathbf{y}_{\mathrm{PL}}$ and $\mathbf{y}_{\mathrm{QL}}$ are the stochastic variations in normal distributions.

The stochastic dynamic of the load is therefore modeled by

$$\mathbf{P}_{\mathrm{L}} = \mathbf{P}_{\mathrm{L0}} + \mathbf{y}_{\mathrm{PL}} \tag{3-3}$$

$$\mathbf{Q}_{\mathrm{L}} = \mathbf{Q}_{\mathrm{L0}} + \mathbf{y}_{\mathrm{QL}} \tag{3-4}$$

where $\mathbf{P}_{\mathrm{L0}}$ and $\mathbf{Q}_{\mathrm{L0}}$ are the mean values of the active and reactive loads, respectively.

Periodicities and autocorrelations have been observed in historical data of loads on the daily basis. However, in the time frame of seconds, loads at different substations have much lower autocorrelations. Refer to [41], this dissertation sets the drifting parameter on the autocorrelations of loads as 0.5 p.u./s.

3.3 Proposed ADM-based Approach for Solving Power System SDEs

3.3.1 Modeling Stochastic Variables

Consider S stochastic variables $y_1(t)$, …, $y_S(t)$, which could be stochastic loads following S different distributions. Each $y_i(t)$ can be transformed by function $g_i(\cdot)$ in (3-5) from some ε_i in a normal distribution. For example, if $y_i(t)$ is a load represented by a normal distribution with certain mean value, then ε_i specifies a zero-mean normal distribution as in (3-7) and $g_i(\cdot)$ shifts it to around the desired mean value like in (3-3) and (3-4).

$$\mathbf{y}(t) = \begin{bmatrix} g_1(\varepsilon_1) & g_2(\varepsilon_2) & \cdots & g_S(\varepsilon_S) \end{bmatrix}^T \tag{3-5}$$

The Ornstein-Uhlenbeck process is utilized to generate each ε_i from (3-6).

$$\dot{\boldsymbol{\varepsilon}}(t) = -\mathbf{a} \circ \boldsymbol{\varepsilon}(t) + \mathbf{b} \circ \mathbf{W}(t) \tag{3-6}$$

where
$$\boldsymbol{\varepsilon}(t) = \begin{bmatrix} \varepsilon_1(t) & \varepsilon_2(t) & \cdots & \varepsilon_S(t) \end{bmatrix}^T$$

$$\mathbf{a}(t) = \begin{bmatrix} a_1(t) & a_2(t) & \cdots & a_S(t) \end{bmatrix}^T$$

$$\mathbf{b}(t) = \begin{bmatrix} b_1(t) & b_2(t) & \cdots & b_S(t) \end{bmatrix}^T$$

$$\varepsilon_i \sim \mathcal{N}(0, b_i^2 / 2a_i) \quad i = 1, 2, \cdots, S \tag{3-7}$$

3.3.2 Solving SDEs Using the ADM

Consider a nonlinear system modeled by SDE (3-8) having M deterministic state variables x_1, …, x_M, such as the state variables of generators, exciters and speed governors, and S stochastic variables y_1, …, y_S.

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t)) \tag{3-8}$$

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) & \cdots & x_M(t) \end{bmatrix}^T$$
$$\mathbf{f}(\cdot) = \begin{bmatrix} f_1(\cdot) & f_2(\cdot) & \cdots & f_M(\cdot) \end{bmatrix}^T$$

To solve $\mathbf{x}(t)$, the procedure in [46] can be used. First, apply Laplace transformation to (3-8) to obtain

$$\mathcal{L}[\mathbf{x}] = \frac{\mathbf{x}(0)}{s} + \frac{\mathcal{L}[\mathbf{f}(\mathbf{x}, \mathbf{y})]}{s} \tag{3-9}$$

Then use (3-11) and (3-12) to calculate the Adomian polynomials under the assumption of (3-10),

$$\mathbf{x}(t) = \sum_{n=0}^{\infty} \mathbf{x}_n(t) \tag{3-10}$$

$$f_k(\mathbf{x}, \mathbf{y}) = \sum_{n=0}^{\infty} A_{k,n}(\mathbf{x}_{0,} \mathbf{x}_{1,} \ldots \mathbf{x}_{n,} \mathbf{y}) \tag{3-11}$$

$$A_{k,n} = \frac{1}{n!} \left[\frac{\partial^n}{\partial \lambda^n} f_k \left(\sum_{k=0}^{n} \mathbf{x}_k \lambda^k, \mathbf{y} \right) \right] \Bigg|_{\lambda=0} \tag{3-12}$$

Recursive formulas (3-13) and (3-14) can be derived by matching terms of $\mathbf{x}(t)$ and $\mathbf{f}(\cdot)$:

$$\mathcal{L}[\mathbf{x}_0] = \mathbf{x}(0)/s \tag{3-13}$$

$$\mathcal{L}[\mathbf{x}_{n+1}] = \mathcal{L}[\mathbf{A}_n]/s \quad n \geq 0 \tag{3-14}$$

where, $\mathbf{A}_n = \left[A_{1,n}, A_{2,n}, \cdots, A_{M,n} \right]^T$

The next step is to apply inverse Laplace transform to both sides of (3-13) and (3-14) to calculate the N-th order SAS of (3-8):

$$\mathbf{x}^{SAS}(t,\mathbf{y}) = \sum_{n=0}^{N} \mathbf{x}_n(t,\mathbf{y}) \tag{3-15}$$

In the resulting SAS, stochastic variables in **y** appear explicitly as symbolic variables.

3.4 Comparison between the Euler-Maruyama Approach and ADM-based Approach

This section applies both the Euler-Maruyama approach and the proposed ADM-based approach to the SMIB system with a stochastic load shown in Fig. 3.2 to illustrate the fundamental difference between the two approaches.

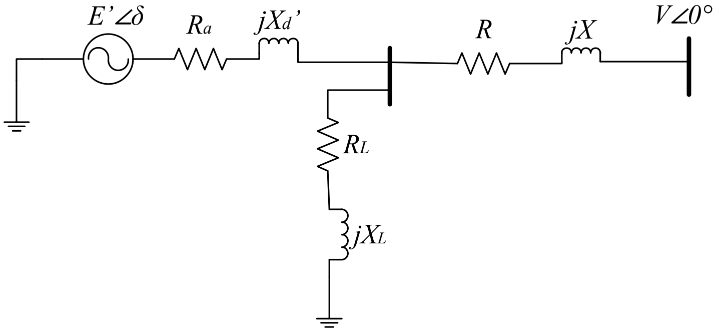


Fig. 3.2. SMIB system with constant impedance load at generator bus.

The stochastic load is connected to the generator bus and has its resistance R_L and reactance X_L modeled by stochastic variables. Thus, the whole system is now modeled by DEs (3-16a), (3-16b) and SDEs (3-16c), (3-16d).

$$\begin{cases} \dot{\delta}=\omega-\omega_R & \text{(3-16a)} \\ \dot{\omega}=\dfrac{\omega_R}{2H}\left(-D\dfrac{\omega-\omega_R}{\omega}+P_m-\left(k_3+\dfrac{E'V}{k_1k_2}\left(k_4\cos(\delta)+k_5\sin(\delta)\right)\right)\right) & \text{(3-16b)} \\ \dot{R}_L=-a_1R_L+b_1W(t) & \text{(3-16c)} \\ \dot{X}_L=-a_2X_L+b_2W(t) & \text{(3-16d)} \end{cases}$$

where,

$$G_L+jB_L=\frac{1}{R_L+jX_L} \tag{3-17a}$$

$$G_S+jB_S=\frac{1}{R_a+jX_d'} \tag{3-17b}$$

$$G_R+jB_R=\frac{1}{R+jX} \tag*{(3-17b)}$$

$$k_1=\frac{(G_L+G_R+G_S)^2}{(B_L+B_R+B_S)}+(B_L+B_R+B_S) \tag{3-17d}$$

$$k_2=(G_L+G_R+G_S)+\frac{(B_L+B_R+B_S)^2}{(G_L+G_R+G_S)} \tag{3-17e}$$

$$k_3=E'^2\left(\frac{G_S(B_L+B_R)+B_S(G_L+G_R)}{k_1}-\frac{B_S(B_L+B_R)-G_S(G_L+G_R)}{k_2}\right) \tag{3-17f}$$

$$k_4=-k_2(B_SG_R+G_SB_R)+k_1(B_SB_R-G_SG_R) \tag{3-17g}$$

$$k_5=-k_2(B_SB_R-G_SG_R)-k_1(B_SG_R+G_SB_R) \tag{3-17h}$$

In (3-17), G_S, B_S, G_R, B_R, G_L, B_L are the conductances and susceptances at the generator sending side, the infinite bus receiving side and the load side, respectively. Since R_L and X_L change stochastically, G_L and B_L can not be treated as constants.

The variances of R_L and X_L depend on the values of drifting parameters a_1 and a_2 and diffusion parameters b_1 and b_2, respectively.

To find the SAS of this system, the first step is to apply ADM to the DEs (3-16a) and (3-16b). The resulting 2nd order SAS for rotor speed ω is,

$$\omega(t) = \sum_{n=0}^{2} \omega_n(t) \tag{3-18}$$

where,

$$\omega_0(t) = \omega(0) \tag{3-19}$$

$$\omega_1(t) = -\frac{t\omega_R}{2H}\left[\frac{D(\omega(0)-\omega_R)}{\omega_R} - P_m + k_3 + \frac{k_4 E'V}{k_1 k_2}\cos(\delta(0)) + \frac{k_5 E'V}{k_1 k_2}\sin(\delta(0))\right] \tag{3-20}$$

$$\begin{aligned}\omega_2(t) = \frac{t^2\omega_R}{8H^2}\Bigg[&\frac{D^2(\omega(0)-\omega_R)}{\omega_R} + D(-P_m + k_3 + \frac{k_4 E'V}{k_1 k_2}\cos(\delta(0)) + \frac{k_5 E'V}{k_1 k_2}\sin(\delta(0)))\\ &+2H\omega_R\frac{k_5 E'V}{k_1 k_2}\cos(\delta(0)) - 2H\omega_R\frac{k_4 E'V}{k_1 k_2}\sin(\delta(0)) - 2H(\omega(0)-\omega_R)\frac{E'V}{k_1 k_2}\cos(\delta(0))\Bigg]\end{aligned} \tag{3-21}$$

Once the SAS of the system's DEs is derived, the SAS of the SDEs can be derived and incorporated into it.

For example, the 2nd order SAS of R_L can be derived using ADM as,

$$R_L(t) = \sum_{n=0}^{2} R_{L,n}(t) \tag{3-22}$$

where,

$$R_{L,0}(t) = R_L(0) + b_1 B(t) \tag{3-23}$$

$$R_{L,1}(t) = -a_1 R_L(0)t - a_1 b_1 \int_0^t B(s_1)ds_1 \tag{3-24}$$

$$R_{L,2}(t) = a_1^2 R_L(0)\frac{t^2}{2!} + a_1^2 b_1 \int_0^t \int_0^{s_1} B(s_2)ds_2 ds_1 \tag{3-25}$$

B(*t*) is the Brownian motion starting at origin and d*B*(*t*)=*W*(*t*)d*t*. Similarly, the 2nd order SAS of X_L is,

$$X_L(t) = \sum_{n=0}^{2} X_{L,n}(t) \tag{3-26}$$

where,

$$X_{L,0}(t) = X_L(0) + b_2 B(t) \tag{3-27}$$

$$X_{L,1}(t) = -a_2 R_L(0)t - a_2 b_2 \int_0^t B(s_1)ds_1 \tag{3-28}$$

$$X_{L,2}(t) = a_2^2 R_L(0)\frac{t^2}{2!} + a_2^2 b_2 \int_0^t \int_0^{s_1} B(s_2)ds_2 ds_1 \tag{3-29}$$

To derive the SAS of the entire system considering both the DEs and SDEs, replace the symbolic variables in the DEs' SAS representing the stochastic variables with the SDEs' SAS, i.e., the 2nd order SAS of the system (3-16) can be derived by replacing the symbolic variables R_L and X_L in (3-18) with SAS (3-22) and (3-26).

For some forms of SDE, the analytical solution may exist. In such cases, the SDEs' analytical solution instead of the SAS also can be incorporated into the DEs' SAS to derive the SAS of the entire system.

For example, the general expression of the SAS terms of (3-16c) can be written as,

$$R_{L,n}(t) = (-1)^n a_1^n R_L(0)\frac{t^n}{n!} + (-1)^n a_1^n b_1 \int_0^t \int_0^{s_1} \cdots \int_0^{s_{n-1}} B(s_n) ds_n \ldots ds_2 ds_1 \quad \text{(3-30)}$$

Therefore the infinite order SAS of (3-16c) is,

$$R_L(t) = R_L(0)\sum_{i=0}^{\infty}\frac{(-a_1 t)^i}{i!} + b_1 B(t) + b_1 \sum_{i=1}^{\infty}(-a_1)^i \int_0^t \int_0^{s_1} \cdots \int_0^{s_{i-1}} B(s_i) ds_i \ldots ds_2 ds_1 \qquad \text{(3-31)}$$

Apply Maclaurin expansion of an exponential function and lemma 2.3 in [50] to (3-31), the solution becomes,

$$R_L(t) = R_L(0)e^{-a_1 t} + b_1 B(t) - a_1 b_1 \int_0^t e^{a_1 s - a_1 t} B(s) ds \qquad \text{(3-32)}$$

Then apply the integration by parts formula,

$$\int_0^t e^{a_1 s} dB(s) = e^{a_1 t} B(t) - \int_0^t a_1 e^{a_1 s} B(s) ds \qquad \text{(3-33)}$$

The close form solution can be found as,

$$R_L(t) = e^{-a_1 t}[R_L(0) + b_1 \int_0^t e^{a_1 s} dB(s)] \qquad \text{(3-34)}$$

In this case the symbolic variable R_L in (3-18) can be replaced by (3-34) instead of (3-22).

On the other hand, for the Euler-Maruyama approach [51][52], since the deterministic model described by (3-16a) and (3-16b) does not permit a close form solution, the sample trajectories of (3-16) have to be numerically computed. The numerical scheme for R_L is shown in (3-35) and the same scheme also applies to X_L.

$$R_{L,n+1}^{(\Delta t)} = R_{L,n}^{(\Delta t)} + a_1 R_{L,n}^{(\Delta t)} \Delta t + b_1 R_{L,n}^{(\Delta t)} \Delta W \qquad \text{(3-35)}$$

In practice the value of ΔW is dependent of the step size Δt for integration.

$$\Delta W \sim \mathcal{N}\ (0, \Delta t^{1/2}) \tag{3-36}$$

3.5 Stability of Stochastic Systems

There are a variety of definitions on the stability of a stochastic dynamical system in literature [54]-[57]. The definition of "asymptotic stability in probability" in [57] can be directly applied to a power system with stochastic variables. That definition is a counterpart of the asymptotic Lyapunov stability of a deterministic system.

Definition 1: Stability in probability

An equilibrium point is said to be *stable in probability* if for given $\mu \in (0, 1)$ and r, there exists $\sigma(\mu,r,t_0) > 0$ such that,

$$P\left\{\sup_{t\geq t_0}\left\|\mathbf{x}(t;x_0,t_0)-\mathbf{x}_{eq}\right\|<r\right\}\geq 1-\mu \tag{3-37}$$

whenever $||\mathbf{x}_0\text{-}\mathbf{x}_{\mathrm{eq}}||<\sigma$.

Definition 2: Asymptotic stability in probability

An equilibrium point is said to be *asymptotic stable in probability* if it is *stable in probability* and for given $\mu \in (0, 1)$, there exists $\sigma_0(\mu)>0$ such that,

$$P\left\{\lim_{t\to\infty}\left[\mathbf{x}(t;x_0,t_0)-\mathbf{x}_{eq}\right]=0\right\}\geq 1-\mu \tag{3-38}$$

whenever $||\mathbf{x}_0\text{-}\mathbf{x}_{\mathrm{eq}}||<\sigma_0$

To analyzed the stability of numerical simulation results, this dissertation modifies (3-38) to (3-39) so that the stability can be accessed using the results of finite time period simulations.

$$P\left\{\left\|\mathbf{x}(t;x_0,t_0)-\mathbf{x}_{eq}\right\|<r_0\right\}\geq 1-\mu \quad \forall t>t_s \tag{3-39}$$

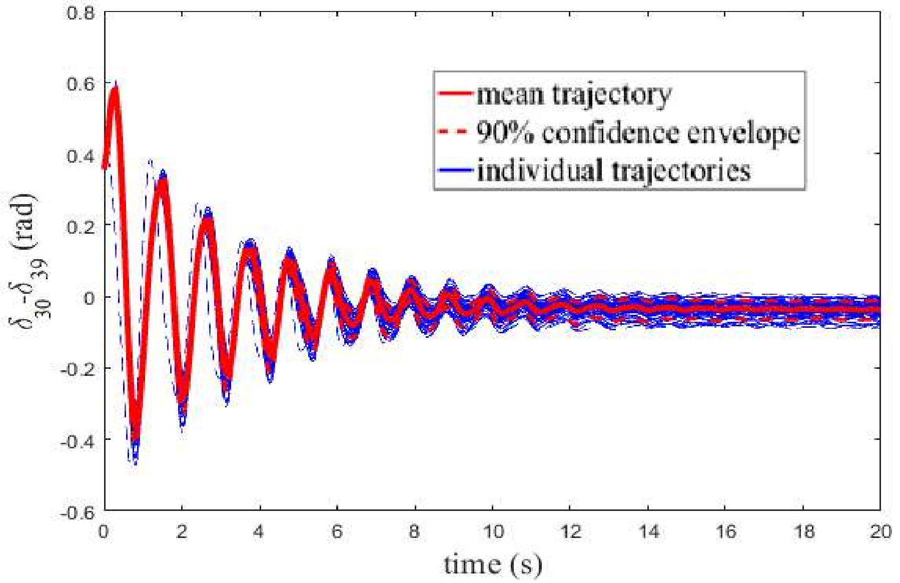
where t_s is a predefined time instant, $\|r_0\|$ is a small positive number.

3.6 Case Studies

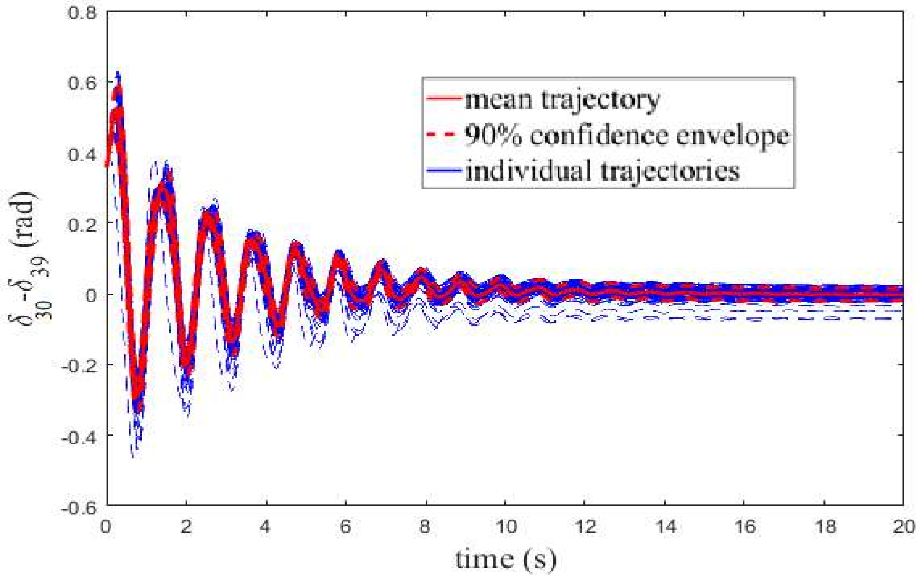
The proposed ADM-based approach is tested on the IEEE 10-machine 39-bus New England system as shown in Fig. 2.8 Selected loads are assumed to change stochastically while all generators are represented by deterministic models. In each case study, the stochastic simulation result by the Euler-Maruyama approach is used as the benchmark, and the 2nd order SASs (i.e. N=2) are used and evaluated every 0.001 s. The value of each stochastic variable is changed every 0.1 s. For each case, 100 sample trajectories are generated. The fault applied in all cases is a 10-cycle 3-phase fault at bus 3 cleared by tripping line 3-4. All simulations are performed in MATLAB R2016a on a desktop computer with an Intel Core i7-3770 3.40GHz CPU and 8 GB RAM.

3.6.1 Stochastic Loads at 5% with Low Variances

In the first case, model the loads at buses 3 and 4 (about 5% of the system load) by Ornstein-Uhlenbeck process. The variances of the loads are 2% of their mean values. The results from the ADM-based approach and the Euler-Maruyama approach are shown in Fig.3.3. Among all the generators, generator 1 has the shortest the electrical distance to bus 3 and 4, hence the rotor angle of it is presented in the following results.



(a) Result from the ADM-based approach.



(b) Result from the Euler-Maruyama approach.

Fig. 3.3. Simulation results of generator 1 rotor angle with loads connecting to bus 3 and 4 represented by stochastic variable with 2% load variation.

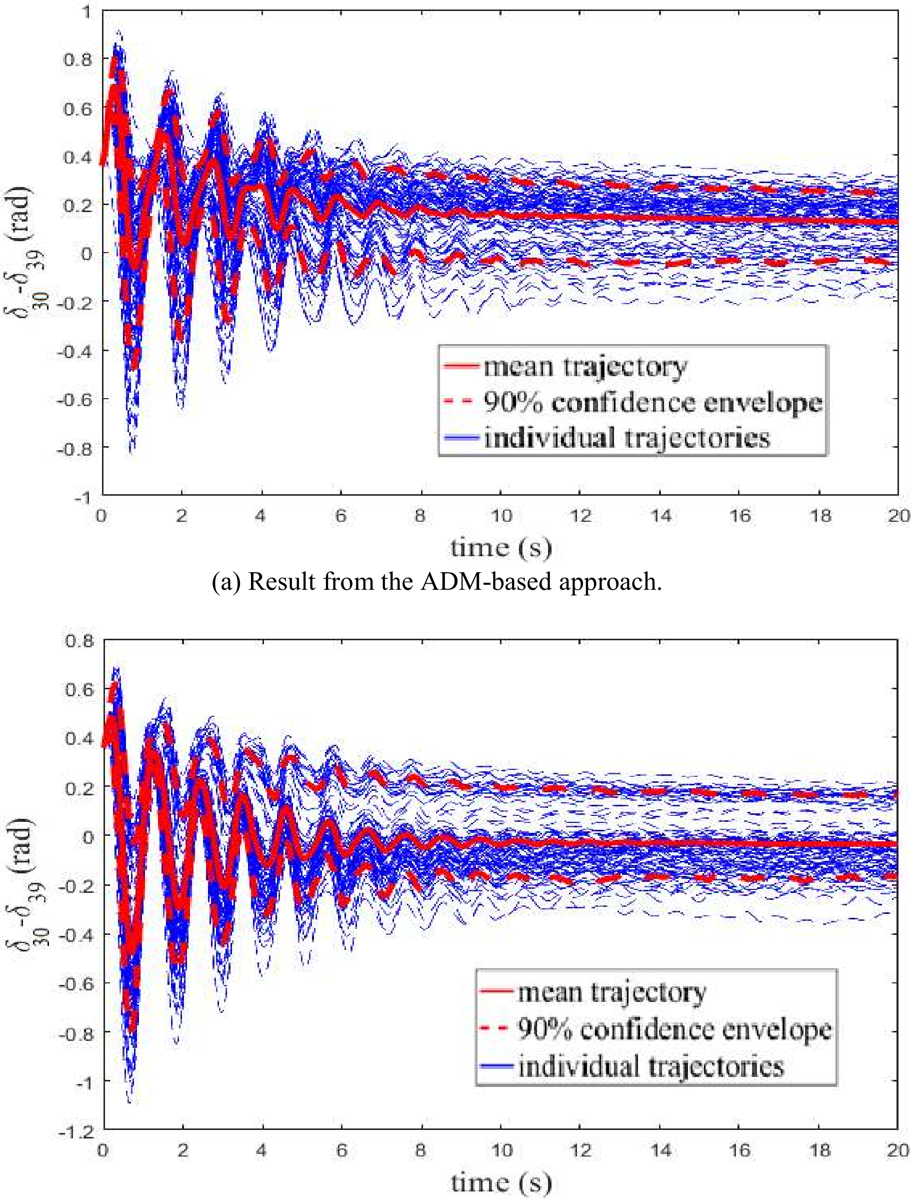
From the simulation results, the deterministic system response is indicated by the mean value and is asymptotically stable. Use the stochastic system stability definition introduced in Section 3.5. When the loads at buses 3 and 4 have small variances, the system behaves similar to a deterministic system, which is asymptotically stable with a probability of 0.9 (t_s=15 s, r_0=0.05 rad/s).

3.6.2 Stochastic Loads at 100% with Low Variances

In the second case, extend stochastic loads to all buses with variances equal to 2% of their mean values. As shown in Fig. 3.4, the simulation results from two approaches agree with each other, which reveal a less stable post-fault system response due to increased uncertainties.

When all the system loads are stochastic, the system is asymptotically stable with a probability of 0.6 (t_s=15 s, r_0=0.05 rad/s). Compared to the first case having only two stochastic loads with the same r_0 value, the probability of the system being asymptotically stable reduces from 0.9 to 0.6.

Therefore, when the percentage of stochastic loads increases, even though the load uncertainties are low and the equilibrium point of the system is almost the same as its deterministic model, the asymptotic stability of the system in probability downgrades. That justifies the necessity of using stochastic load models to study the stability of power systems with a high penetration of stochastic loads.



(a) Result from the ADM-based approach.

(b) Result from the Euler-Maruyama approach.

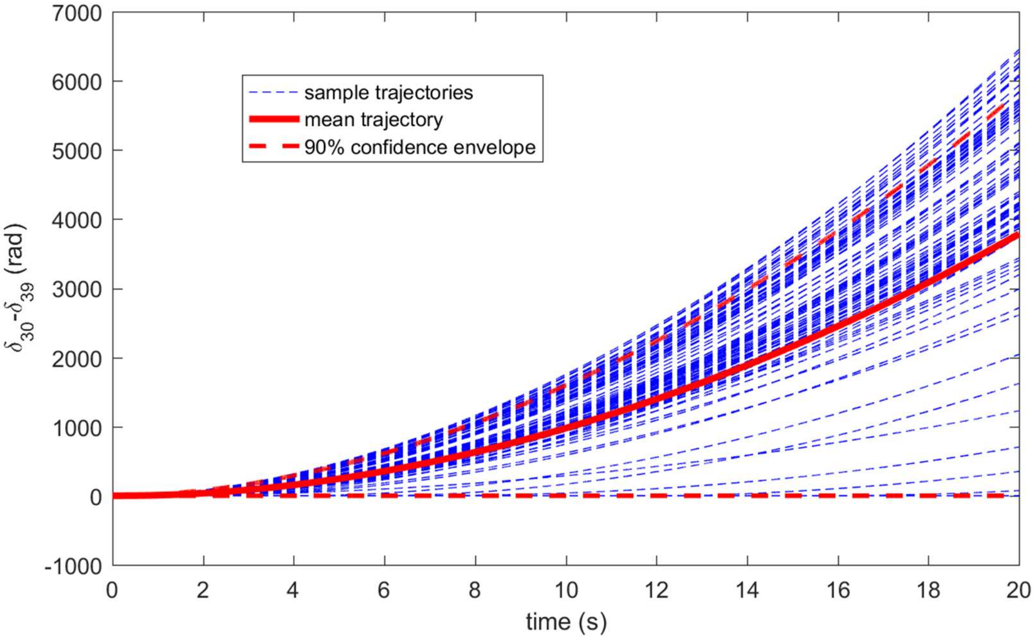
Fig. 3.4. Simulation results of generator 1 rotor angle with all loads represented by stochastic variable with 2% load variation.

3.6.3 Stochastic Loads at 100% with High Variances

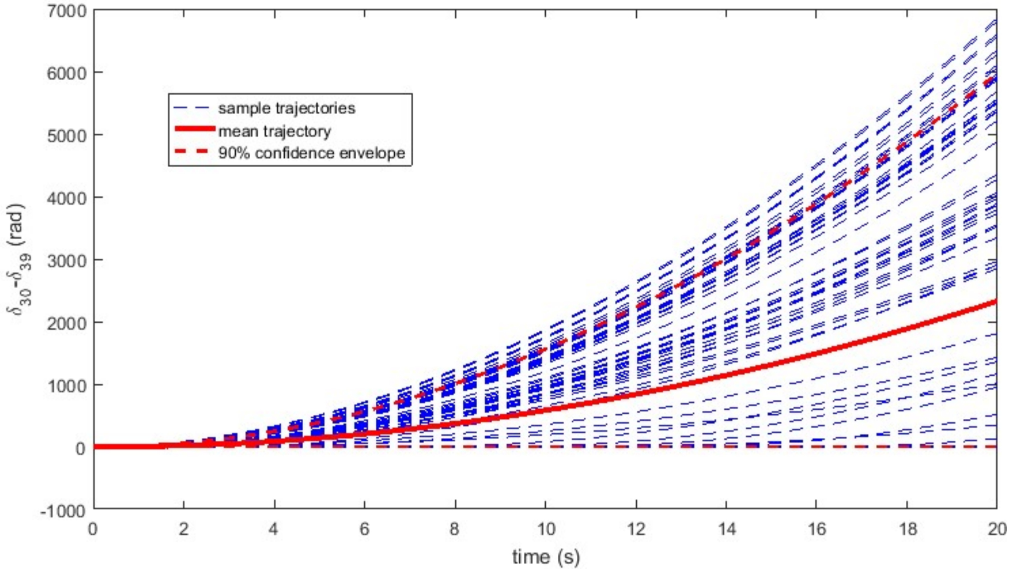
In the third case, all the loads are represented by stochastic loads and the variances of the loads are increased to 4% of the mean values. This case may represent a scenario having DERs widely deployed in distribution networks, which make the aggregated bus load seen from each transmission or sub-transmission substation behave more stochastically. The simulation results from the ADM-based approach and Euler-Maruyama approach are shown in Fig. 3.5.

The ADM-based approach agrees with the Euler-Maruyama approach on the simulation results. Both of them show that the system loses its stability when the variance of the loads increases to 4% of their mean values. The instability is due to the cumulative effect of stochastic load variations. The 90% confidence envelope can be utilized as an indicator of the system stability. Unlike Fig. 3.4, the 90% confidence envelope in Fig. 3.5 is not bounded any more, indicating a 0.9 probability of the system losing stability.

Bus voltages also reflect the impact from high load uncertainties as shown in Fig. 3.6 about the voltage magnitude of bus 30, denoted by V_{30}. With loads of high uncertainties, the system has an increased risk of under- and over-voltage issues because the imbalance between generation and load is magnified by increased load uncertainties. That also indicates the importance of stochastic power system simulation when penetration of DERs becomes high.

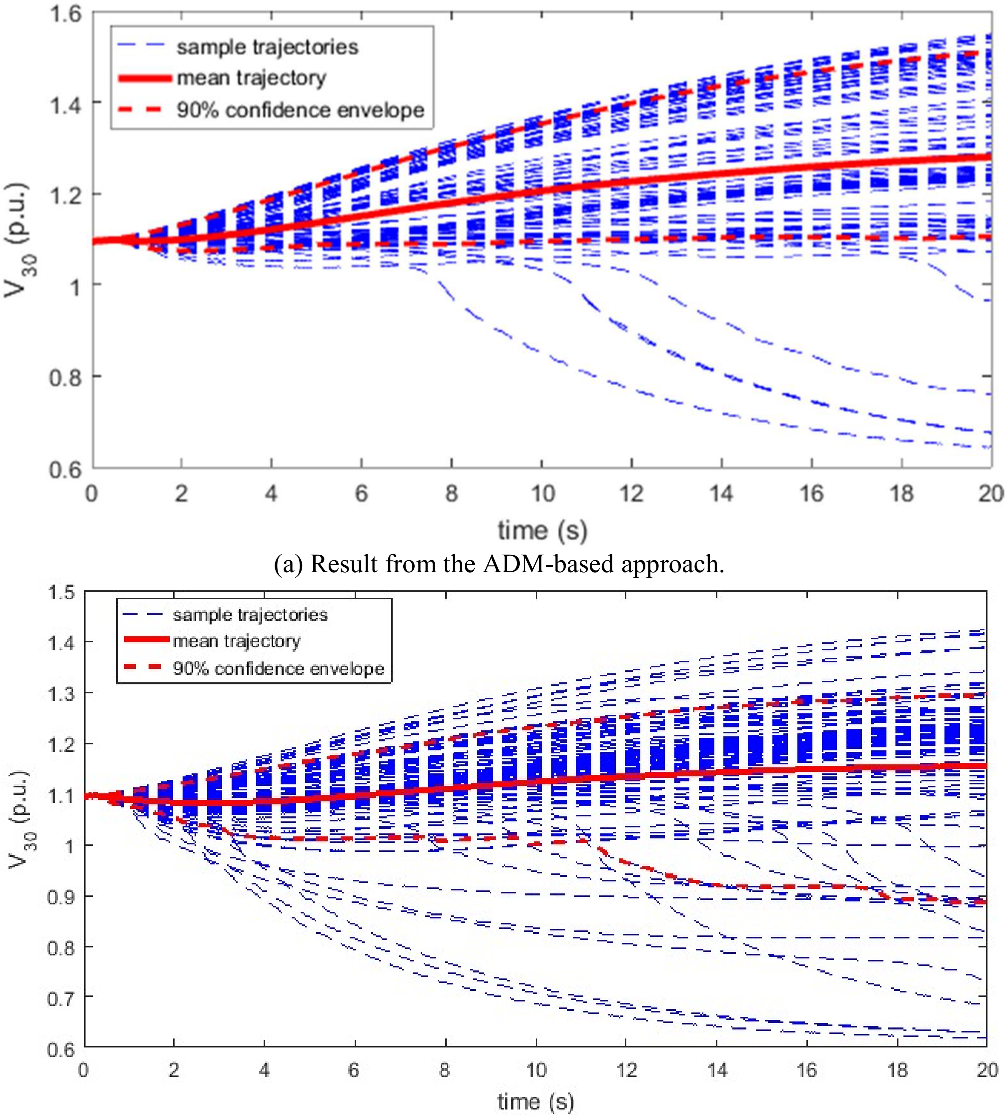


(a) Result from the ADM-based approach.



(a) Result from the Euler-Maruyama approach.

Fig. 3.5. Simulation results of generator 1 rotor angle with all loads represented by stochastic variable with 4% load variation.

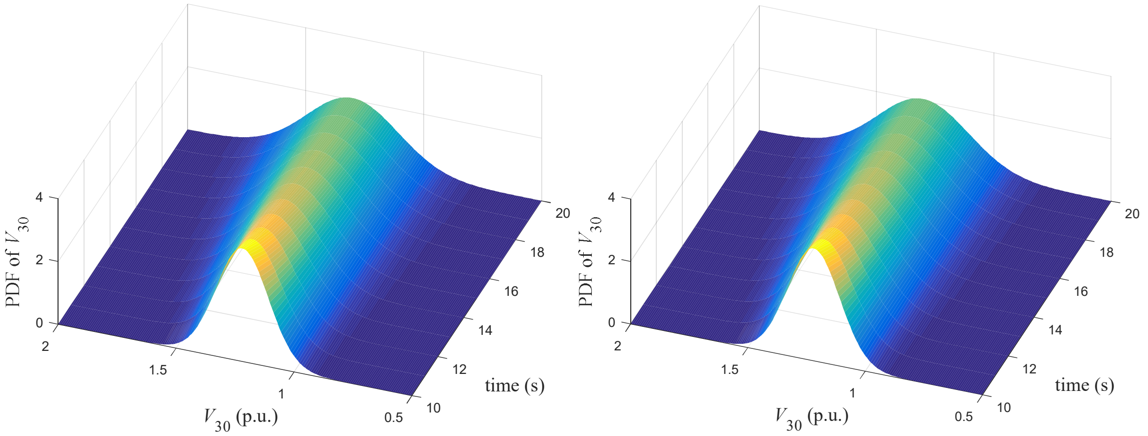


(a) Result from the ADM-based approach.

(b) Result from the Euler-Maruyama approach.

Fig. 3.6. Simulation results of bus voltage at bus 30 with all loads represented by stochastic variable with 4% load variation.

From results of stochastic power system simulation, how the probability distribution function (PDF) of a system variable evolves in time during a post-contingency period can be estimated and fit into an anticipated probability distribution for analysis. As an example, if we assume V_{30} to follow a normal distribution at each time instant with the mean value and variance varying with time, Fig. 3.7 shows the evolutions of its PDF using simulation results from both the ADM-based approach and Euler-Maruyama approach for comparison. Fig. 3.7a basically matches Fig. 3.7b, indicating the accuracy of the proposed ADM-based approach in reflecting the evaluation of the PDF. From Fig. 3.7, as time elapses, the PDF of the bus voltage not only shifts the mean value but also increases the variance indicated by the increasing width of the shape. Such information is not available from deterministic power system simulation. The longer the system is subjected to the effect of stochastic variables the bigger variance and larger uncertainty the system has in post-contingency dynamics.



(a) ADM-based approach.

(b) Euler-Maruyama approach.

Fig. 3.7. Evolution of the pdf of the voltage magnitude at bus 30 from t=10 s to 20 s.

3.6.4 Variances of State Variables

To compare the accuracy of the numerical results from the ADM-based approach and Euler-Maruyama approach, the mean value and standard deviation of the 100 trajectories are compared. For case A, as shown in Fig. 3.8 and Fig. 3.9, the ADM-based approach achieves comparable accuracy as the Euler-Maruyama approach in terms of both mean value and standard deviation value.

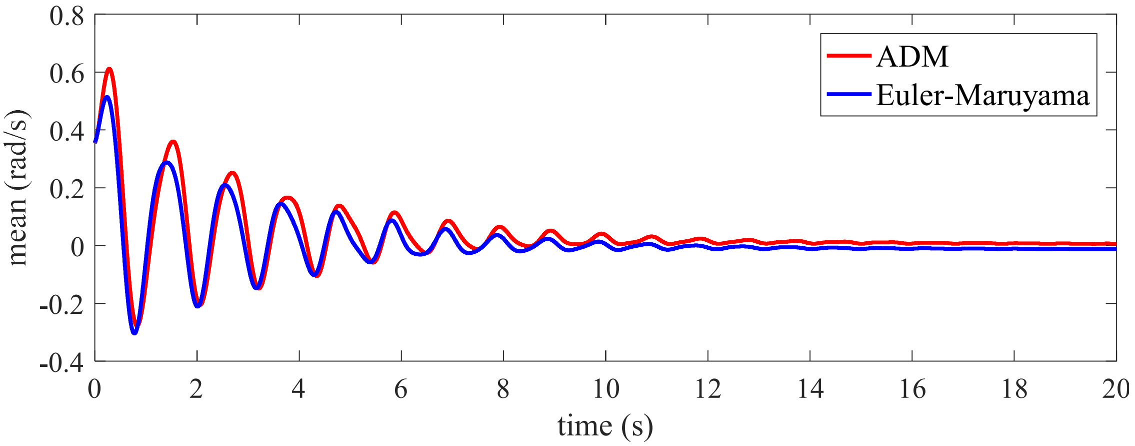


Fig. 3.8. Mean value of generator 1's rotor angle for case A.

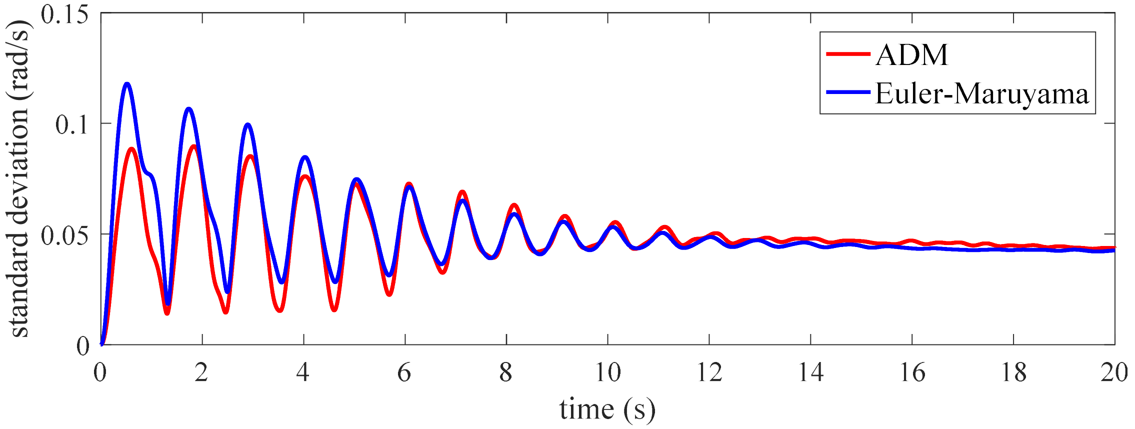


Fig. 3.9. Standard deviation of generator 1's rotor angle for case A.

As more loads are modeled as stochastic, the variance of state variables grows accordingly. The mean value and standard deviation of the rotor angle of generator 1 for case B are shown in Fig. 3.10 and Fig. 3.11. In case B, the standard deviation reaches its largest value 0.25 rad/s during the first swing, which is larger than the largest standard deviation in case A, 0.1 rad/s.

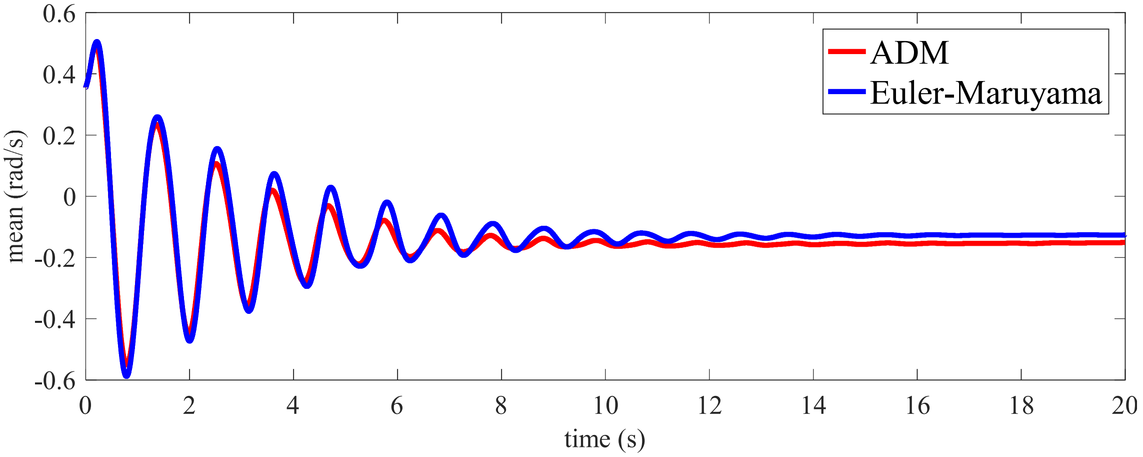


Fig. 3.10. Mean value of generator 1's rotor angle for case B.

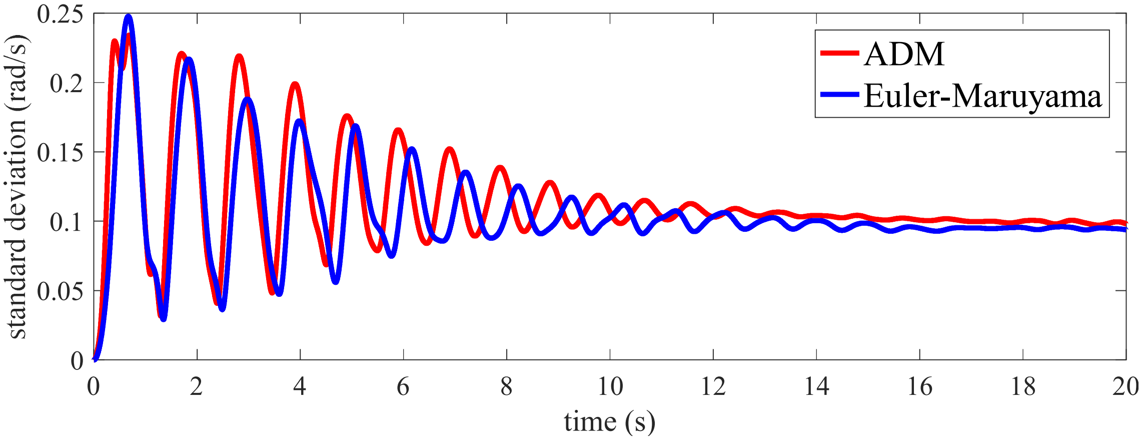


Fig. 3.11. Standard deviation of generator 1's rotor angle for case B.

3.6.5 Comparison on Time Performance

The time performances for cases A, B and C of the ADM-based approach and Euler-Maruyama approach are compared in Table 3.1, from which the ADM-based approach takes less than 50% of the time cost of the Euler-Maruyama approach. The advantage of the ADM-based approach in time performance is more prominent when many simulation runs are required. As discussed in [21], the ADM-based approach is inherently suitable for parallel implementation, which could help further improve the time performance if high-performance parallel computers are available.

Table 3.1. Time performance on the 4th-Order model System.

Time costs (s)	Stochastic loads at all buses (Case B, C)	Stochastic loads at buses 3 and 4 (Case A)
Ito calculus single run	11.6	11.4
Ito calculus 100 runs	1165.1	1142.4
ADM single run	5.1	5.1
ADM 100 runs	511. 0	503.6

3.7 Conclusion

This chapter proposes an alternative approach for stochastic simulation of power systems. Using the SAS derived from the ADM, the stochastic effects from load uncertainties can be taken into considerations. The result from the proposed approach is benchmarked with that from the Euler-Maruyama approach. Since the evaluation of SASs is faster than the integration with the Euler-Maruyama approach, the proposed approach has an obviously advantage in time performance. This is critical when a large number of simulation runs need to be performed for simulating stochastic behaviors of a future power grid having a high penetration of DERs. The simulation results on different levels of stochastic loads show that when the level of load uncertainty is low, the deterministic simulation is still trustworthy compared to the mean-value trajectory from stochastic simulation, but, once the level of load uncertainty becomes high, the mean-value trajectory no longer represents the true behavior of the system.

CHAPTER FOUR

PARAREAL IN TIME POWER SYSTEM SIMULATION

Recent rapid advances of smart grid technologies have begun to change the landscape of the power industry and give rise to a far more dynamically controlled grid. Furthermore, power systems become increasingly uncertain as the penetration of intermittent renewable sources such as wind and solar increases. There is a need for real-time, online action which requires faster-than-real-time computation and decision making. The Parareal method, along with other advances in parallel and high performance computing, promises to give answer to this challenging problem.

4.1 Parareal Application to Power System

The Parareal algorithm includes two layers of solution propagations. The first layer is the approximate and fast (coarse) solution propagation. It is computed both before the fine solution propagation and during each correction of the fine solution. The coarse solver gives a less accurate but computationally cheaper solution to guide the fine solver to the true solution.

$$Coarse: \tilde{U}_n = C_{\Delta T}(T_{n-1}, U_{n-1}, \Delta T) \tag{4-1}$$

where $\tilde{U}_n$ is the system state obtained from coarse solver at time T_n, and ΔT is the coarse solver integration step-size.

The second layer is the accurate (fine) solver correction. For fine solver, more detailed integrators and smaller integration increments are used.

$$Fine: \hat{U}_n = F_{\Delta t}(T_{n-1}, U_{n-1}, \Delta t) \tag{4-2}$$

where $\hat{U}_n$ is the system state obtained from the fine solver at time T_n, and Δt is the fine solver's integration increment.

Initially, the coarse solver first propagates the coarse solution through the entire simulation interval.

$$\begin{aligned} &U_n^0 = \tilde{U}_n^0 = G_{\Delta T}(U_{n-1}^0) \\ &\forall 1 \le n < N \\ &U_0^1 = U_0^0 \end{aligned} \tag{4-3}$$

where the superscript denotes the iteration.

The fine solver corrects the initial guess provided by the coarse solver in an iterative way. In each iteration, the fine solution propagates over each coarse interval. The error between the new fine solution and the last coarse solution is computed.

$$\hat{U}_n^k = F_{\Delta t}(U_{n-1}^{k-1}) \ \ \forall 1 \le n < N \tag{4-4}$$

$$\Delta_n^k = \hat{U}_n^k - \tilde{U}_n^{k-1} \ \ \forall 1 \le n < N \tag{4-5}$$

where N is the number of intervals for the fine solver.

Then the error is utilized to update the coarse solution. After the update, the coarse solver propagates a new coarse solution corresponding to those updated values.

$$\begin{aligned} &U_n^k = \tilde{U}_n^k + \Delta_n^k \\ &where\, \tilde{U}_n^k = C_{\Delta T}(U_{n-1}^k) \end{aligned} \tag{4-6}$$

After *M* iterations, the first *M* coarse intervals are corrected to the true solution. If it takes the total number of coarse solutions $N_{coarse} = N$ iterations to converge, the Parareal algorithm will have the same speed as the traditional sequential computation. If it takes $N_{iteration} < N_{coarse}$ to converge, the theoretical speedup [58] of the Parareal approach is,

$$Speedup_{\text{theoretical}} = \frac{N_{\text{coarse}}}{N_{\text{iteration}}} \tag{4-7}$$

However, this speedup is based on the assumption that the coarse solver time to update and propagate is negligible, which is the ideal case. In practice, the actual speedup is,

$$Speedup_{\text{actual}} = \frac{CPU\ Time_{\text{sequential algorithm}}}{CPU\ Time_{\text{parallel algorithm}}} \tag{4-8}$$

Choices for a coarse solver can be based on the following:

1) Using a larger time increment than that for the fine solver;
2) Using a time-stepping method faster and less accurate than that for the fine solver;
3) Using a reduced model;
4) Limiting the total number of iterations in a coarse solver.

In previous studies [59], the combination of 1) and 4) has been tested. For 2), some other fast time domain simulation technique [60] has the potential of improving the Parareal algorithm performance. Choice 3) has not been investigated in power systems problems and is the subject of this dissertation.

The coarse solver utilized in this dissertation is the Midpoint-Trapezoidal predictor-corrector method.

$$y_{n+1}^{(k)} = y_n + hf(t_n + \frac{h}{2}, y_n + \frac{h}{2} f(t_n, y_n)) \tag{4-9}$$

$$y_{n+1}^{(k+1)} = y_n + \frac{h}{2}\left[f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{(k)}) \right] \tag{4-10}$$

The Midpoint-Trapezoidal predictor-corrector method is an implicit method. In order to achieve accurate solutions, many iterations may be required. However, since the accuracy of coarse solution is not of a major concern, only one iteration is computed.

The fine solver is the Runge-Kutta 4th order (R-K 4) integration method.

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_1) \\ k_3 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_2) \\ k_2 &= f(t_n + h, y_n + hk_3) \\ y_{n+1} &= y_n + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4] \end{aligned} \tag{4-11}$$

The step-size of the R-K 4 fine solver is chosen to be much smaller than that of the Midpoint-Trapezoidal coarse solver.

Even though previous results in [59] show that there is a speedup with such a coarse solver configuration, there is still a large gap between the actual and theoretical speedups. In order to improve the performance, the time consumed by the coarse solver has to be as short as possible. The next sub-section investigates how to simplify the underlying physics and modeling of the synchronous generators and apply a reduced model for the coarse solver to achieve a time performance closer to the theoretical speedup.

4.2 Simplified Generator Model

The synchronous machines in this sub-section are modeled as IEEE Model 2.2 shown in Fig. 4.1. On the d-axis, there are one damper winding and the field winding, while on the q-axis, there are two damper windings. Sub-transient saliency is considered using a dummy coil approach [61]. There are 9 state variables for this generator model listed in (4-12)-(4-20):

$$\frac{d\delta}{dt} = \omega_B S_m \tag{4-12}$$

$$2H\frac{dS_m}{dt} = T_m - T_e - DS_m \tag{4-13}$$

$$\frac{d\psi_f}{dt} = -\frac{\omega_B R_f}{x_{fl}}\psi_f + \frac{\omega_B R_f}{x_{fl}}\psi_{ad} + \frac{\omega_B R_f}{x_{ad}}E_{fd} \tag{4-14}$$

$$\frac{d\psi_g}{dt} = -\frac{\omega_B R_g}{x_{gl}}\psi_g + \frac{\omega_B R_g}{x_{gl}}\psi_{aq} \tag{4-15}$$

$$\frac{d\psi_h}{dt} = -\frac{\omega_B R_h}{x_{hl}}\psi_h + \frac{\omega_B R_h}{x_{hl}}\psi_{ad} \tag{4-16}$$

$$\frac{d\psi_k}{dt} = -\frac{\omega_B R_k}{x_{kl}}\psi_k + \frac{\omega_B R_k}{x_{kl}}\psi_{aq} \tag{4-17}$$

$$\frac{dE''_{dummy}}{dt} = \frac{1}{T_{dummy}}\left[-E''_{dummy} - (x''_{qs} - x''_{ds})i_q\right] \qquad (4\text{-}18)$$

$$\frac{d\hat{x}''_{ads}}{dt} = \frac{1}{T_1}\left[-\hat{x}''_{ads} + F_s(V_g, I_g, x_{ad})\right] \qquad (4\text{-}19)$$

$$\frac{d\hat{x}''_{aqs}}{dt} = \frac{1}{T_1}\left[-\hat{x}''_{aqs} + F_s(V_g, I_g, x_{aq})\right] \qquad (4\text{-}20)$$

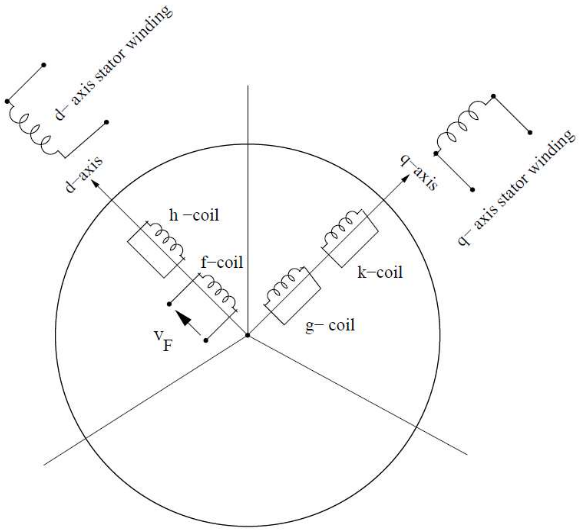


Fig. 4.1. 2.2 model of a synchronous machine.

where: δ is rotor angle; S_m is speed slip; *H* is inertia constant of a generator; *D* is mechanical damping; ω_B is base speed; T_m is mechanical input torque; T_e is electromagnetic torque; ψ_f is the flux linkage of field winding; ψ_h is the flux linkage of damper winding along d-axis; ψ_g and ψ_k are the flux linkages of the two damper windings along q-axis; ψ_{ad}

is d-axis mutual flux linkage; ψ_{aq} is q-axis mutual flux linkage; E_{fd} is generator field voltage; R_f, R_h, R_g, R_k are the resistances of each coil; x_{fl}, x_{hl}, x_{gl}, x_{kl} are the leakage reactances of each coil; x_{ad} is d-axis mutual synchronous reactance; E''_{dummy} is a fictitious voltage source of the dummy coil; T_{dummy} is the open circuit time constant of the dummy coil; x''_{ds}, x''_{qs} are the saturated sub-transient reactances of d-axis and q-axis, respectively; i_q is q-axis winding current; $\hat{x}''_{ads}$ and $\hat{x}''_{aqs}$ are the saturated sub-transient mutual reactances of d-axis and q-axis, respectively.

More information about IEEE 2.2 generator model can be found in [61]. The complete model including the IEEE type 1 excitation system and the 1st order turbine-governor models has 15 states variables for each generator. However, if only (4-12)-(4-15) are considered, the generator model can be reduced to the IEEE Model 1.1, which is a 4th order model. By equating the left hand side of (4-16)-(4-20) to zero, the 2.2 model is converted to the 1.1 type model and (4-21)-(4-25) are turned into the constraints of the 5th to 9th state variables' values. Those values are updated using the solution of the algebraic equations ignoring related fast dynamics for every integration step.

$$\psi_h = \psi_{ad} \tag{4-21}$$

$$\psi_k = \psi_{aq} \tag{4-22}$$

$$E''_{dummy} = -(x''_{qs} - x''_{ds})i_q \tag{4-23}$$

$$\hat{x}''_{ads} = F_s(V_g, I_g, x_{ad}) = \frac{1}{\frac{1}{x_{ad}} + \frac{1}{x_{fl}} + \frac{1}{x_{hl}}} \tag{4-24}$$

$$\hat{x}''_{aqs} = F_s(V_g, I_g, x_{aq}) = \frac{1}{\dfrac{1}{x_{aqs}} + \dfrac{1}{x_{gl}} + \dfrac{1}{x_{kl}}} \tag{4-25}$$

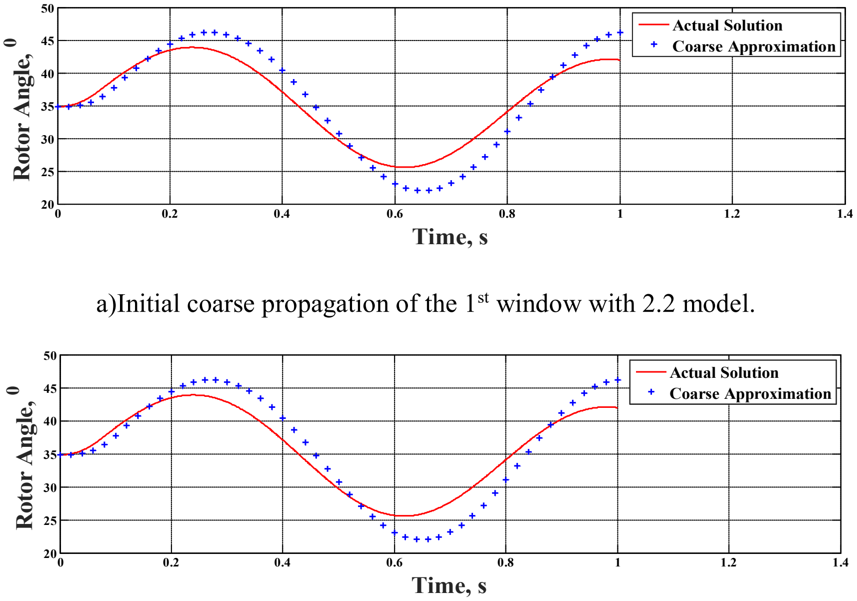
Equations (4-16)-(4-20) model the eddy current, the sub-transient process and the relaxation of sub-transient saliency in the generators, and their dynamics are deemed to be already damped out within each coarse interval. Therefore, it is reasonable to substitute the differential equations (4-16)-(4-20) with algebraic equations (4-21)-(4-25) for the coarse solver to simplify the model and reduce the computational burden of the coarse solution propagation. But the obvious question is whether this assumption will negatively impact the convergence performance of the Parareal algorithm.

4.3 Simplified Generator Model Case Study

Three different test cases have been studied to validate the speedup of applying a simplified model for the coarse solver. They are the IEEE 3-generator 9-bus system, the IEEE New England 10-generator 39-bus system and the IEEE 327-generator 2383-bus Polish system. The test platform is a workstation with Intel Xeon E5-2650, 16-core processor. Ideal parallelism is assumed. (i.e. each fine interval is computed in an individual processor, and the communication overhead between processors are neglected.) In all 3 cases, there are 500 coarse intervals covering a 10 s simulation period. It has been reported in [59] that dividing the simulation into windows can reduce the required iteration number and overall execution time despite introducing sequential computations. Therefore, the simulation is divided in ten 1 s windows each containing 50 coarse intervals. Within each coarse interval, there are 20 fine intervals. Due to space constraints, results for single severe disturbances are shown in each system, representative of performances for other possible contingencies.

4.3.1 IEEE 3-generator 9-bus system

A 3-phase fault on bus 5 is applied and cleared after 4 cycles. As shown in Fig. 4.2, there is no significant difference between the coarse solutions using the 2.2 model and 1.1 model.



a)Initial coarse propagation of the 1st window with 2.2 model.

b)Initial coarse propagation of the 1st window with 1.1 model.

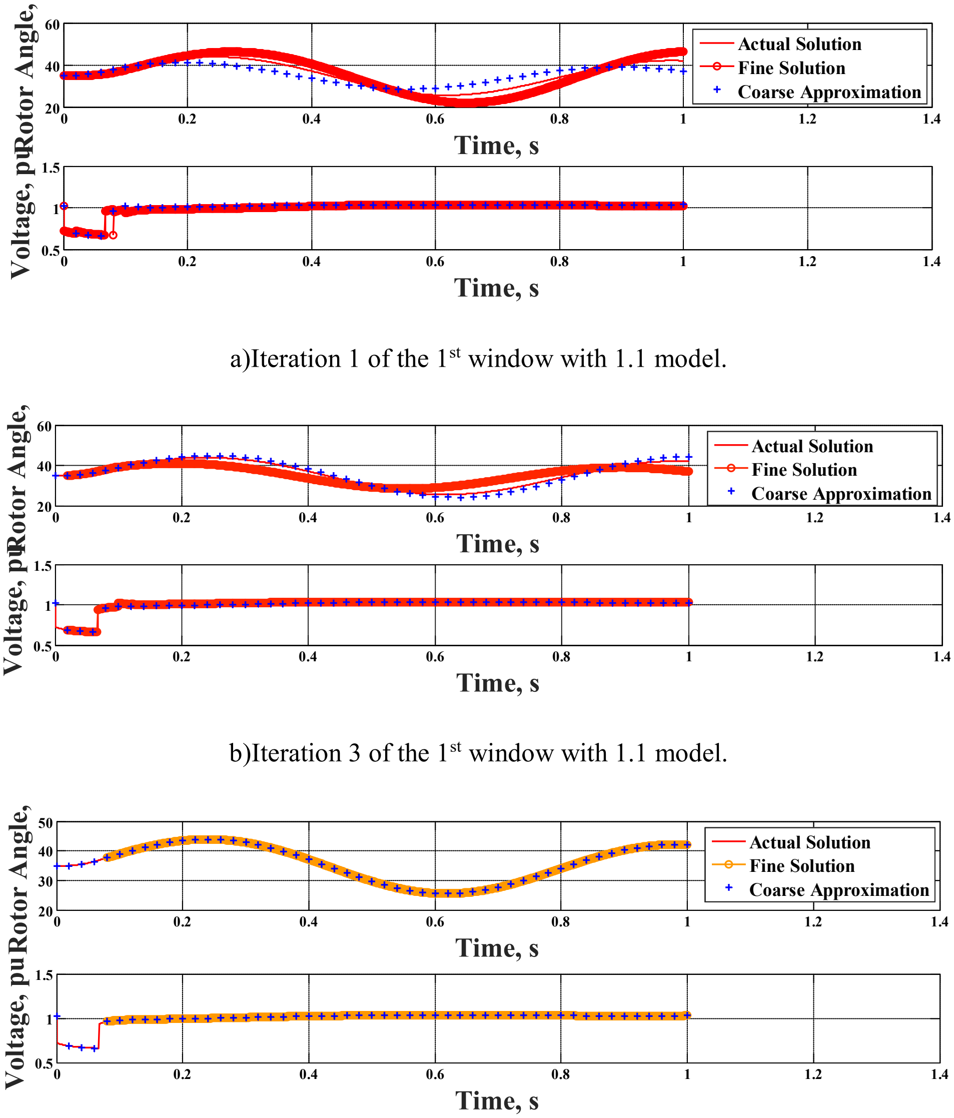
Fig. 4.2. Initial coarse solution comparison of 2.2 and 1.1 model.

The time performances are listed in TABLE 4.1. A considerable percentage of the total simulation time is taken by the coarse solver. When the 1.1 model is utilized, the coarse solver still takes 1.6050 s, but it is 0.0417 s shorter than the 2.2 model. The coarse simulation is reduced by 3%. For a larger system, this percentage is expected to be larger, because the more generators a system has, the fewer differential equations will have to be integrated.

Table 4.1. IEEE 3-Generator 9-Bus System Time Performance.

Coarse Solver	**2.2 model**	**1.1 model**
Total Parareal time	2.2439 s	2.1959 s
Coarse time	1.6467 s	1.6050 s
Coarse time reduced	3%	
Sequential time	13.1381 s	
Actual speed up	5.8550	5.9830

More importantly, the coarse solution with 1.1 model takes the same number of iterations to converge to the fine solution as 2.2 model does. Fig. 4.3 shows the process of the coarse solution converging to the true solution after 5 iterations.



a)Iteration 1 of the 1st window with 1.1 model.

b)Iteration 3 of the 1st window with 1.1 model.

c)Iteration 5 of the 1st window with 1.1 model.

Fig. 4.3. Iterations of 1st window with 1.1 model.

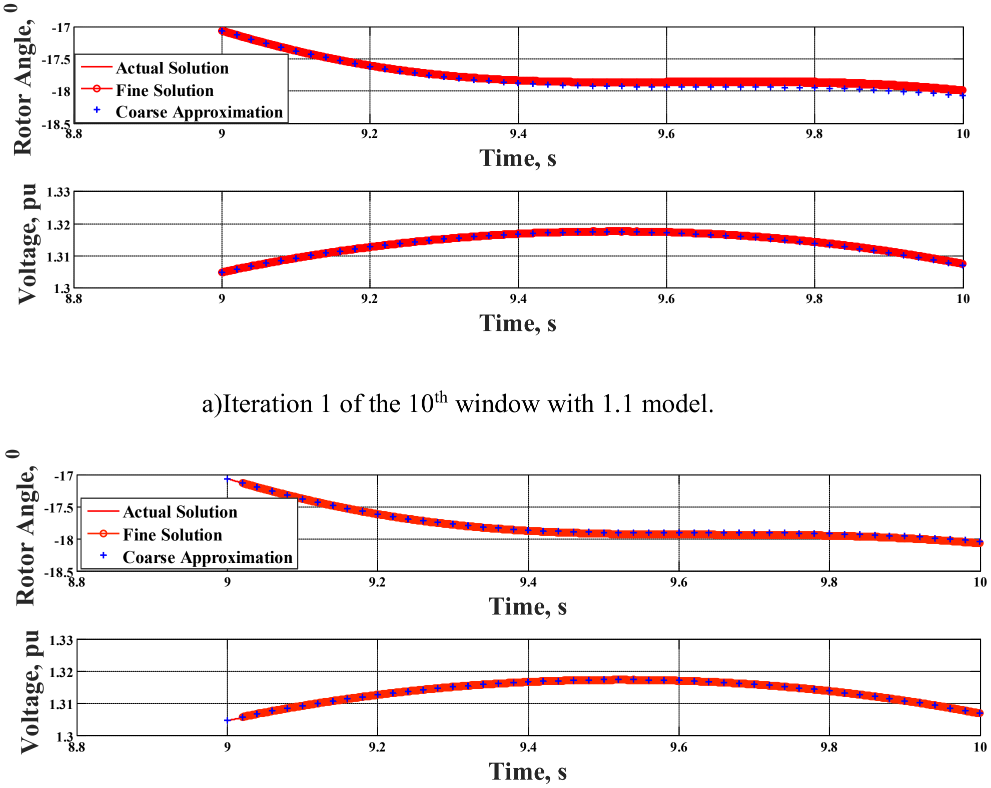
4.3.2 IEEE New England 10-generator 39-bus system

A temporary 3-phase fault on the line between bus 11 and bus 2 is applied and disappears after 4 cycles without any line tripping. As with the IEEE 3-generator 9-bus system, after replacing the generator model in the coarse solver by the 1.1 model, the accuracy of the solution is not affected. As shown in Fig. 4.4, the coarse solution with the 1.1 model converges to true solution after 2 iterations for window No. 5, which is the same number of iterations taken by using the 2.2 model.

Table 4.2 shows the time performances for this system. The percentage of the reduced coarse time is similar to that for the IEEE 3-generator 10-bus system, because the system size is not significantly larger as indicated by the sequential computation times in TABLE 4.1 and TABLE 4.2. There is not a big difference between these two systems. The potential of using the reduced models in the coarse solver is truly tested on the following, much bigger system.

Table 4.2. IEEE 10-Generator 39-Bus System Time Performance.

Coarse Solver	**2.2 model**	**1.1 model**
Total Parareal time	2.2544 s	2.2230 s
Coarse time	1.6617 s	1.6348 s
Coarse time reduced	2%	
Sequential time	13.7689 s	
Actual speed up	6.1075	6.1938



a)Iteration 1 of the 10th window with 1.1 model.

b)Iteration 2 of the 10th window with 1.1 model.

Fig. 4.4. Iterations of 10th window with 1.1 model.

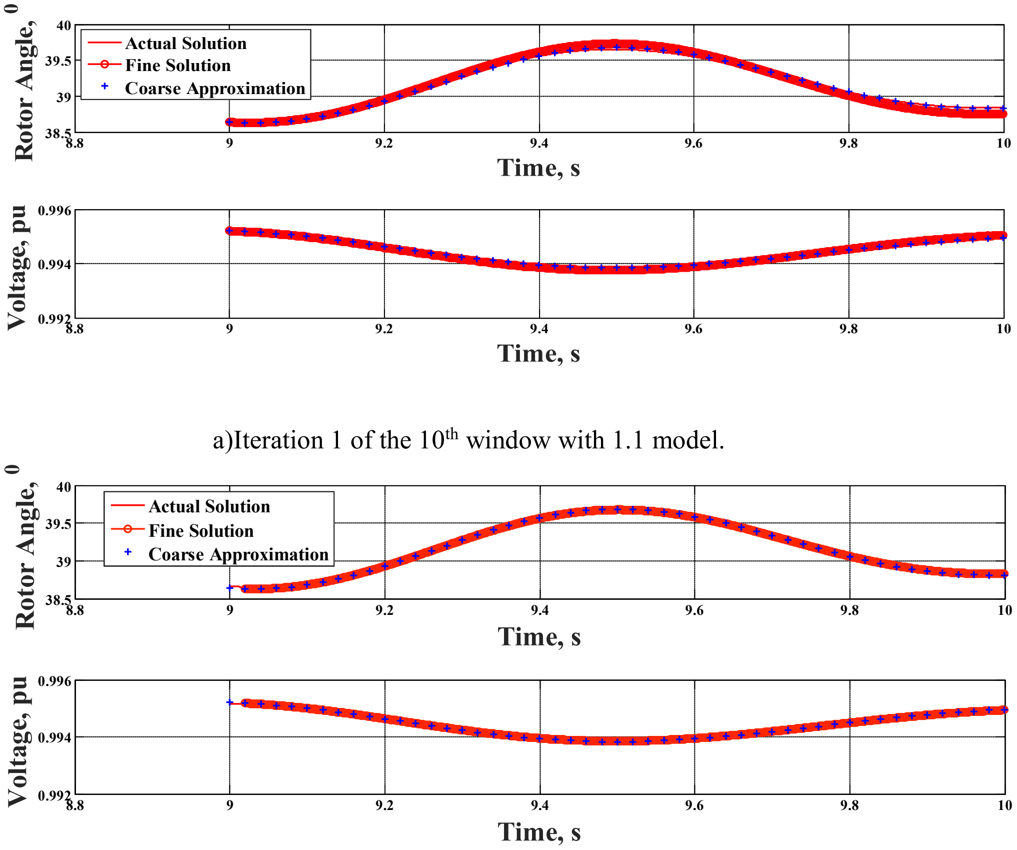
4.3.3 IEEE 327-generator 2382-bus Polish system

A 3-phase bus fault is applied to bus 11 and cleared after 4 cycles. Same as with the smaller systems, the simulation accuracy is not influenced by the simplification of the generator model for the coarse solver through the entire simulation period as shown in Fig. 4.5.

The time performances are listed in Table 4.3. For this system with 327 generators, the coarse time is significantly reduced. The 14% reduction of coarse time gives the Parareal algorithm 13% improvement of speedup. Although the coarse time still takes almost half of the total simulation time, the idea of applying a reduced model for the coarse solver to improve the time performance of the Parareal algorithm is validated and is expected to scale up when solving larger systems.

Table 4.3. IEEE 327-Generator 2383-Bus Polish System Time Performance.

Coarse Solver	**2.2 model**	**1.1 model**
Total Parareal time	82.3150 s	72.6367 s
Coarse time	49.4318 s	43.3946 s
Coarse time reduced	14%	
Sequential time	682.0455 s	
Actual speed up	8.2858	9.3898



a)Iteration 1 of the 10th window with 1.1 model.

b)Iteration 2 of the 10th window with 1.1 model.

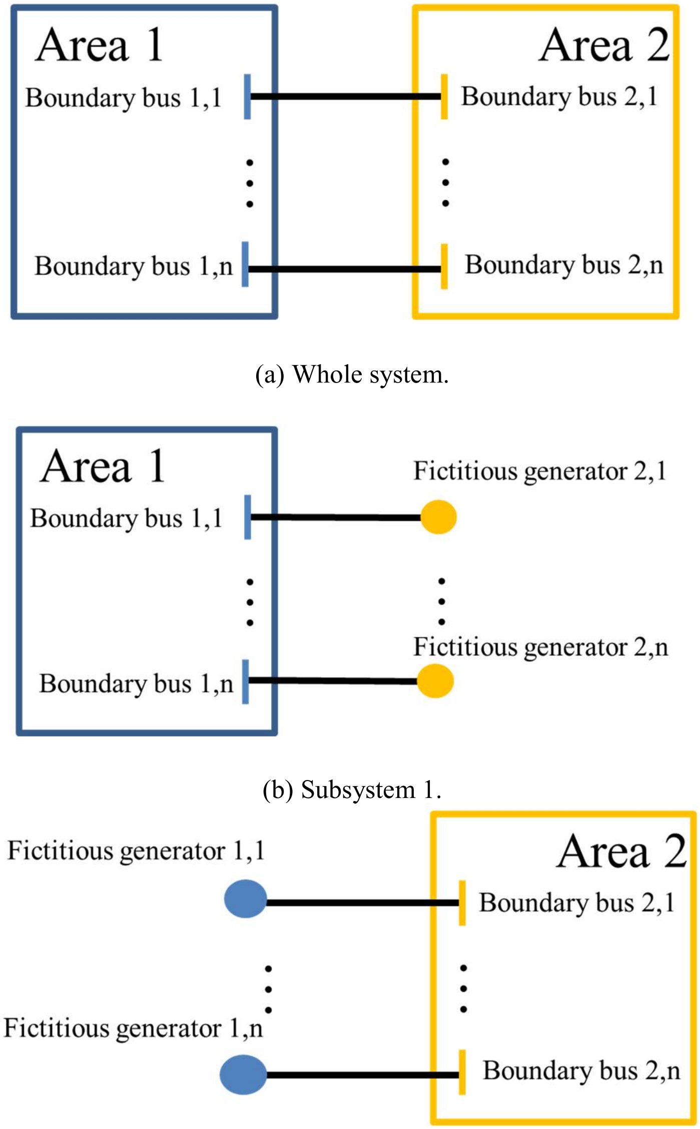
Fig. 4.5. Iterations of 10th window with 1.1 model.

4.4 Embedding Spatial Decomposition in Parareal in Time Algorithm

The Parareal in time approach provides only one dimension of decomposition – temporal decomposition. To further improve the execution time performance of the numerical integration process of power system dynamic simulation, another dimension of decomposition, spatial decomposition, can be considered.

Spatial decomposition can be interpreted in its literal meaning here, as a separation of the system into two or more areas. In the simplest case, the system is typically divided between a study area and an external area. The numerical integration of each area is carried out simultaneously but, instead of treating them as two independent systems, information has to be exchanged at the boundary (interface) of these two areas to maintain the accuracy of the solution of the entire system. The solution at the boundary is achieved by considering the input from both areas. This calculation is done at every numerical integration step, thus keeping the integrity of the whole system even though each area is being simulated separately in parallel. If the system is decomposed into *N* areas and the time for boundary synchronization is negligible, then the theoretical speedup of the parallel simulation would be *N* times faster than simulation of the whole system.

The spatial decomposition in this dissertation follows the approach proposed in [62]. Given a whole system shown in Fig. 4.6a, the two individual subsystems can be defined as shown in Fig. 4.6b and Fig. 4.6c, with fictitious generators representing the rest of the system attached to the boundary buses.



(a) Whole system.

(b) Subsystem 1.

(c) Subsystem 2.

Fig. 4.6. Spatial decomposition that decomposes the whole system into 2 subsystems.

The spatial decomposition procedure is implemented as shown in Fig. 4.7. Each subsystem is first simulated to find their respective solutions for a certain time period. When a preset time for matching up the boundary is reached, the two subsystems will exchange the information of the voltage phasors at the boundary buses and solve (4-26) using Newton's method as proposed in [62].

$$\begin{cases} \mathbf{V}_{bb,1} = \mathbf{V}_{fg,1}(\mathbf{V}_{bb,2}) \\ \mathbf{V}_{bb,2} = \mathbf{V}_{fg,2}(\mathbf{V}_{bb,1}) \end{cases} \tag{4-26}$$

Where $\mathbf{V}_{\mathrm{bb},1}$, $\mathbf{V}_{\mathrm{bb},2}$ are the vectors comprised of all voltage phasors of the boundary buses in areas 1 and 2, respectively; $\mathbf{V}_{\mathrm{fg},1}$ and $\mathbf{V}_{\mathrm{fg},2}$ are the vectors comprised of all voltage phasors of the fictitious generators in subsystem 2 and 1 respectively. After several iterations, $\mathbf{V}_{\mathrm{bb},1}$ and $\mathbf{V}_{\mathrm{bb},2}$ are solved as the boundary bus values for the next time step.

This spatial decomposition structure allows the parallelism across different subsystems. However, the frequency of boundary bus information exchange is the key to maintaining an acceptable accuracy. These spatial information exchanges have to be properly designed to be compatible with the temporal information exchanges introduced by Parareal in time algorithm.

4.4.1 Initial Coarse Propagation

This step tolerates a simpler solver and longer coarse time increment for integration. The propagation along time axis is sequential with optional parallelization of spatial areas.

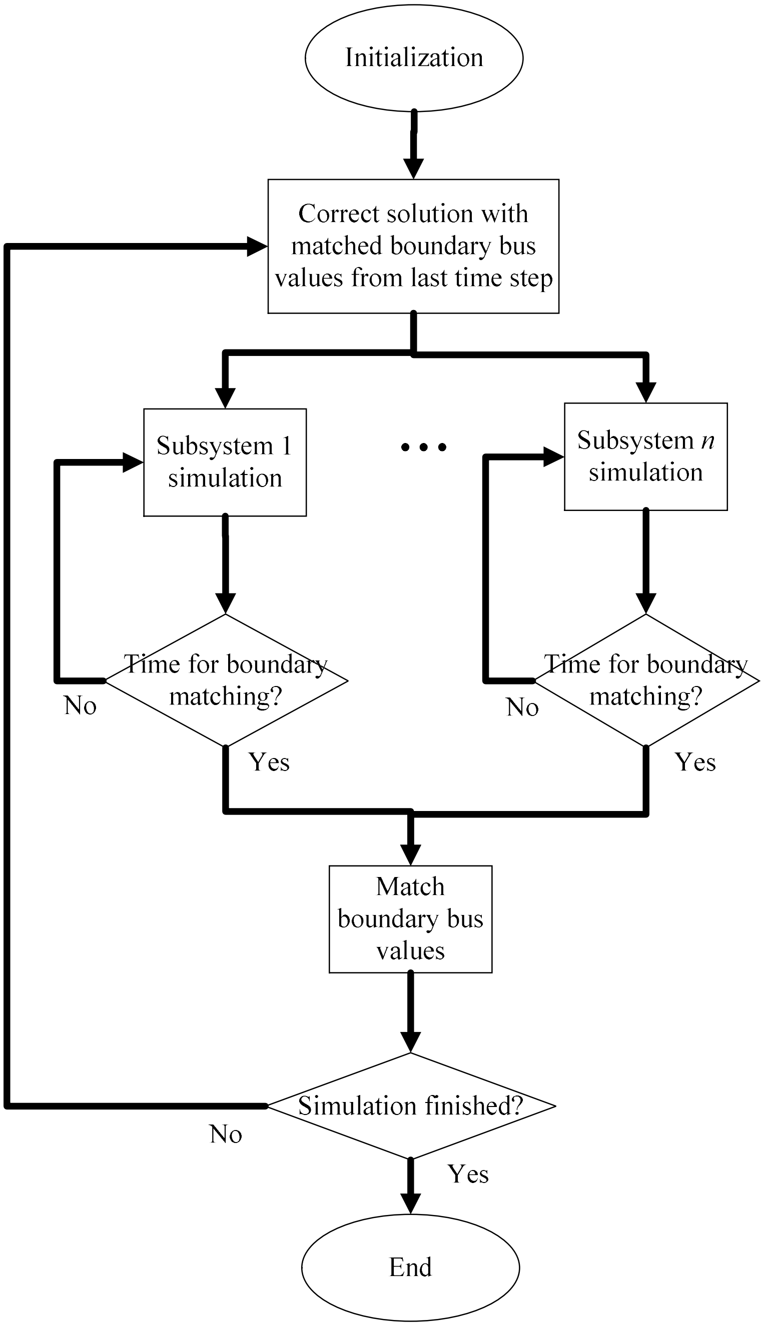

Initialization
Correct solution with matched boundary bus values from last time step
Subsystem 1 simulation
• • •
Subsystem *n* simulation
Time for boundary matching?
No
Yes
Time for boundary matching?
No
Yes
Match boundary bus values
Simulation finished?
No
Yes
End

Fig. 4.7. Spatial decomposition simulation procedure.

Since in the expression for the theoretical speedup of the Parareal algorithm it is assumed that the initial coarse propagation takes no time (compared to the overall required time), it is preferable to simulate different areas in parallel to make the actual speedup closer to the theoretical. However, the larger coarse interval size delays the boundary information exchanges between different areas. Therefore, the inaccuracy of the initial coarse solution is due to not only the simpler solver and large integration steps (coarse intervals), but also the delayed boundary information exchange. The extra inaccuracy caused by the last may become “the final drop” that leads to the divergence of later coarse propagations. Therefore, in this dissertation, the initial coarse propagation is sequential both spatially and temporally.

4.4.2 Fine Propagation

The fine propagation function has two levels of parallelism. The first level is the parallelism within the coarse intervals. For each iteration, the fine solution is propagated within all the coarse intervals simultaneously. The second level is the parallelism between the different spatial areas within each coarse interval. In the second level, unlike the typical spatial decomposition approach, the boundary synchronization is skipped, which can minimize the communication between processors for the entire fine propagation function.

The essence of both spatial decomposition method and temporal decomposition method is to, first, separate the entire task into independent tasks to yield a set of less accurate solutions, and, then, correct them by exchanging information between tasks.

For the Parareal in time approach, the first step is designed to be the fine propagation and the second step is the coarse propagation. In the former, the communication between tasks is limited to the minimum. The correction of the mismatches between tasks is left for coarse propagation. To integrate spatial decomposition with temporal decomposition, the same principle is followed. For each fine interval, the mismatch between areas is kept without correction. This design sacrifices the accuracy of the fine solution for independency of tasks. The loss of accuracy is temporary as the mismatches between areas (spatial) and coarse intervals (temporal) are corrected all together in the coarse propagation function later.

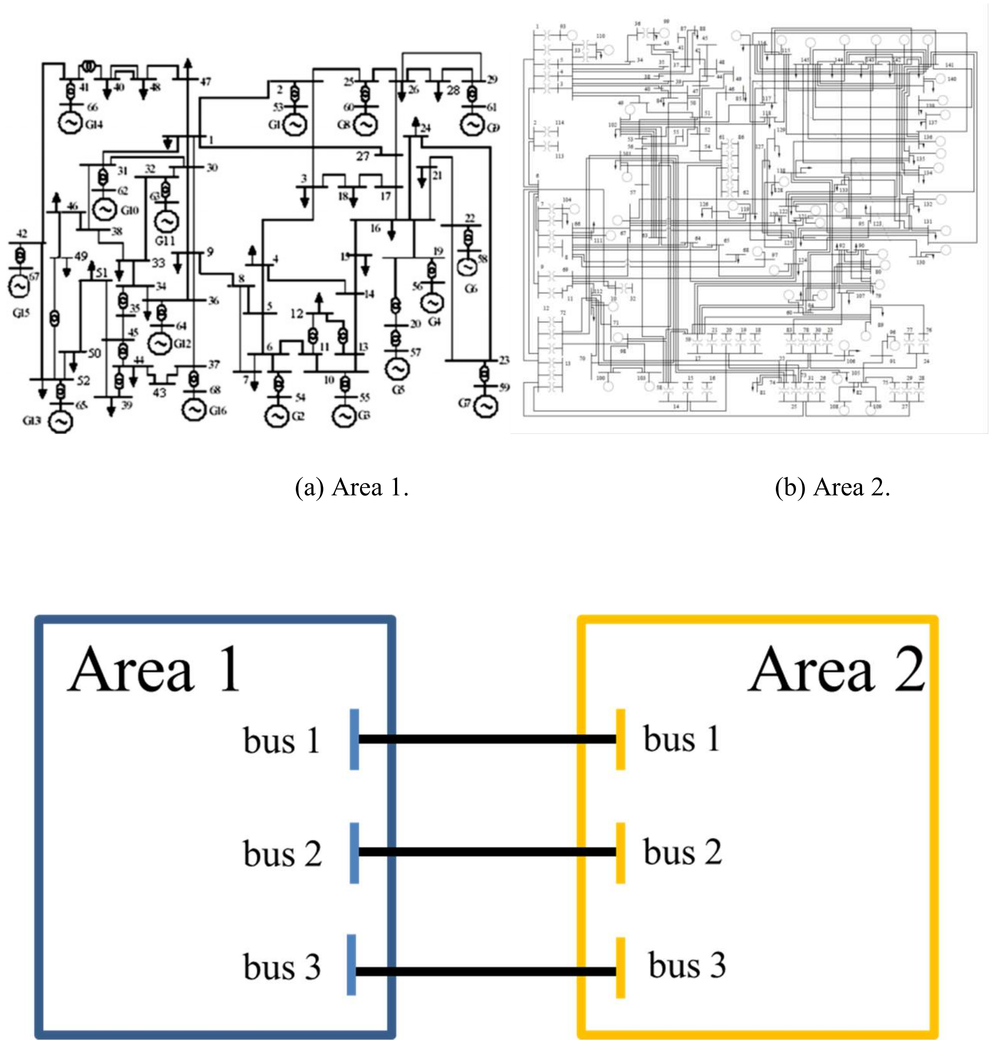
4.4.3 Coarse Propagation

The coarse propagation function is temporally sequential, because it needs to link all the coarse intervals along time axis and correct the mismatch. The spatial parallelism is optional in this step depending on the complexity of the system. The boundary information exchange is integrated into this function to correct the spatial mismatch. In terms of accuracy, either propagating the coarse solution from different areas simultaneously or one by one gives the same solution. At the first glance, the spatial parallelism should be a preferable choice. But after taking a closer look at the structure of the coarse propagation, it is not difficult to notice that the tasks of simulating different areas are not as independent as in the fine propagation. If the spatial parallelism is implemented, all processors have to communicate for each integration step (coarse interval). This means the task assigned for each processor is just one step of integration of one area (in contrast, in the case of fine propagation each processor has N_f fine steps of N

areas to compute), which could be too trivial a task to assign an individual processor to. When the task for each processor is too simple to run, the communication overhead may outweigh the computational time gain and the parallelism may become impractical. However, if the system is very large, one step of integration of one area is already a heavy task, in which case it is still worth to implement spatial parallelism in the coarse propagation function. In this dissertation, since the system is relatively small, the iterative coarse propagation is sequential both spatially and temporally.

4.5 Case Study for Spatial Temporal Decomposition

The test system is formed by connecting the IEEE 16-machine 68-bus system (area 1 as study) to the IEEE 50-machine 145-bus system (area 2 as external) through buses 1, 2, 3, respectively, in each area shown in Fig. 4.8. The fault is applied at bus 6 in area 1 and cleared by tripping the line 6-11 at the near and remote ends after 0.05s and 0.1s, respectively. The same test system has been used in [62]. The coarse interval is set at 0.01 s for pre-fault and during-fault simulation, and 0.03 s for after fault simulation. The fine interval is set as 0.001 s, for pre-fault and during-fault simulation, and 0.003 s for after fault simulation. The 15 s simulation period is comprised of 512 coarse intervals, each coarse comprised of 10 fine intervals. The solution converges in 2 iterations.



(a) Area 1.

(b) Area 2.

c) Whole test system.

Fig. 4.8. Test system.

The simulation results of all generator rotor angles in the study area are shown in Fig. 4.9. The simulation results from the proposed approach agree with those from the sequential simulation of the whole system.

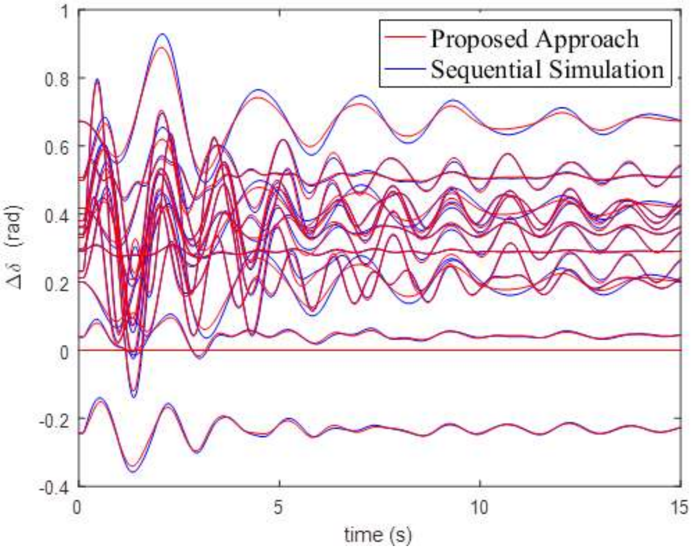


Fig. 4.9. Comparison of relative rotor angles (machine 13 in area 1 as reference) in area 1 from sequential simulation (blue curves) and spatial-temporal decomposition (red curves).

As shown in Table 4.4, using the combined spatial-temporal decomposition Parareal approach can achieve a 12.2 times theoretical speedup. However, Fig. 4.10 shows that the majority of the computing time is comprised of initial and iterative course propagations. This means there is still room left for further improvement in the performance of this approach.

Table 4.4. Time Performance of Spatial Temporal Decomposition Parareal.

	Spatial-Temporal Decomposition Parareal	Temporal Decomposition Parareal
Initial Coarse Time	6.1685 s	
Fine Time	0.6331 s	0.9556 s
Fine Time Speedup	1.5094	
Coarse Time	13.3120	
Total Parareal Time	20.1136 s	20.4361
Sequential Time	245.7169 s	
Spatial-Temporal Decomposition Parareal Speedup	12.2165	12.0237

The comparison of time performance for the 15 s simulation between Parareal with spatial decomposition and Parareal only is shown in Table 4.4. The former is broken down between functional steps in Fig. 4.10.

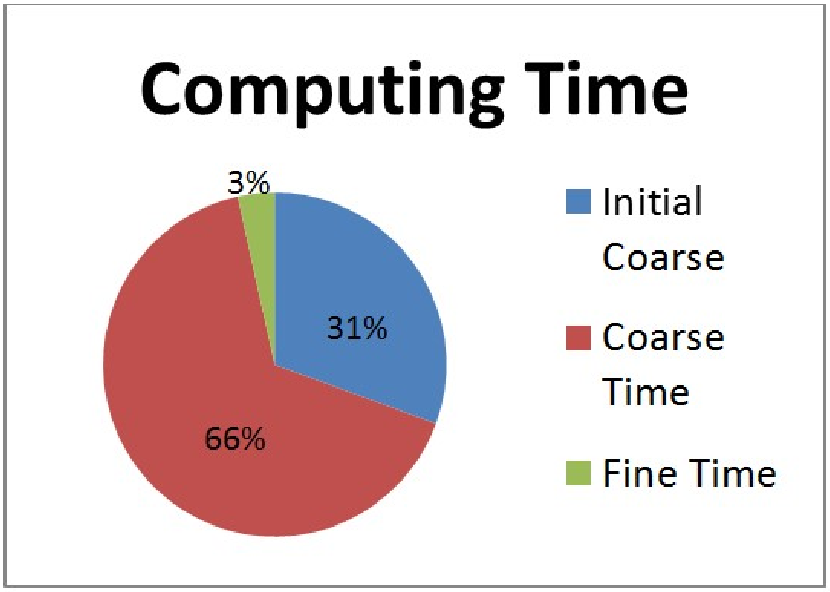


Fig. 4.10. Different steps and their percentage in the total computing time.

To test the generality of the spatial decomposition, a fault that excites inter-area mode has also been studied. Using the small signal analysis tool from power system toolbox (PST) [23], a 3-phase fault on bus 19 in area 1, cleared by tripping the line 19-20 at the near and remote ends after 0.05s and 0.1s, respectively, is identified to be able to excite a 0.7 Hz oscillation between machine 3 in area 2 and machines 4, 6, 7 and 9 in area 1. The compass plot of the rotor angle terms of the inter-area mode eigenvector is shown in Fig. 4.11.

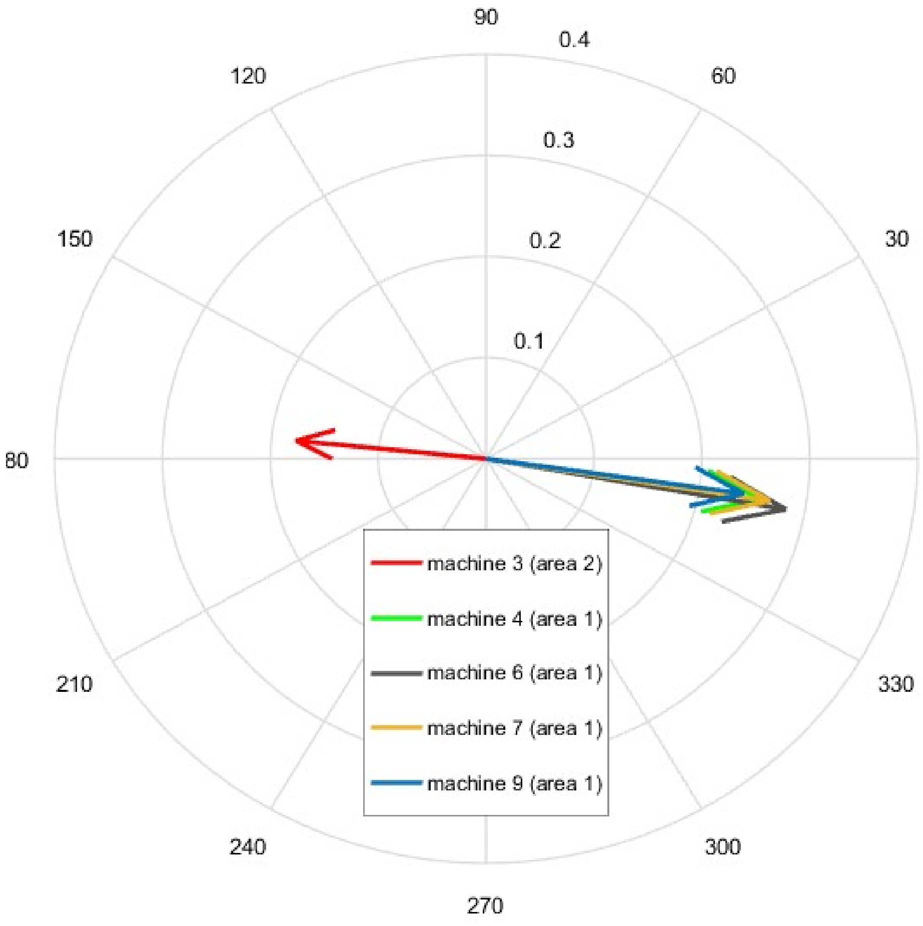


Fig. 4.11. Compass plot of rotor angle terms of the inter-area mode eigenvector.

The simulation results for this case are shown in Fig. 4.12. Again, the proposed approach is as accurate as the sequential simulation of the whole system. This confirms the validity of the proposed approach that it can provide accurate simulation results for not only the contingencies that excite local oscillation modes in certain area, but also the contingencies that excite inter-area oscillations and may be challenging to capture in the decomposed system.

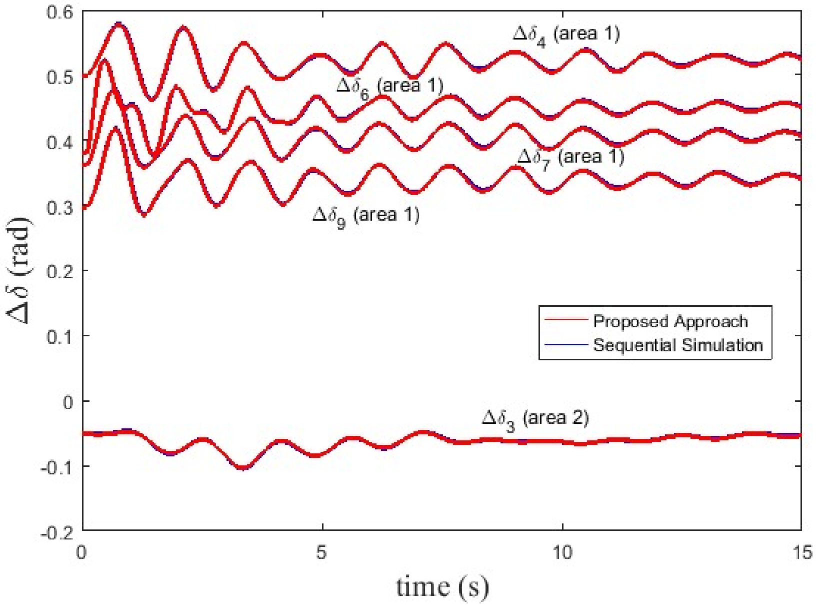


Fig. 4.12. Comparison of relative rotor angles (machine 13 in area 1 as reference) of machines 4, 6, 7 and 9 in area 1 and machine 3 in area 2 from sequential simulation (blue curves) and from spatial-temporal decomposition (red curves).

4.6 Test on EI system

The basic parareal algorithm is tested on the US Eastern Interconnection (EI). A modification regarding to the network solution efficiency is made to accommodate the large number of buses. Also a MATLAB base graphic user interface (GUI) is building to facilitate the choice of continence and the analysis of the simulation results.

4.6.1 EI System Overview

The system data is based on the 2014 summer data in PSSE format. Since the parareal solver is coded in MATLAB and it calls Matpower as its power flow solver, the following modification has been made to the data.

- Extracted the largest island in the raw data. (Ignore part of the Canadian system)
- Renumbered the bus number in sequential order.
- Fixed several incorrect machine base values in raw data.
- Converted the data from PSSE format to Matpower format.
- Converted 3-winding transformers to equivalent 2-winding transformers.
- Converted sectional lines to equivalent lines with fictitious middle buses.
- Converted switchable shunts to fix shunts after reaching power flow solution.
- Matched up the DC lines' real power transfers at inverter and rectifier buses to eliminate DC lines.
- Created the dynamic data in Parareal format using machine MVA bases.
- Fine-tuned the exciter and governor output limits to ensure reasonable no-fault simulation results in Parareal.

The fine-tuned EI system has the statistics listed in Table 4.5

Table 4.5. EI system statistics.

Device	Number
Bus	70285
Line	85639
Generator	5617

4.6.2 Solving the Network Equation

The Parareal algorithm uses a partitioned approach to solve the DEs and AEs alternatively during the simulation. For every integration step, the network voltages **V** need to be recalculated considering the updated current injections **I** from generator buses.

$$\mathbf{YV} = \mathbf{I} \tag{4-27}$$

where **Y** is the admittance matrix of the system. In the EI system case, it is a 70285 by 70285 sparse matrix. The location of non-zero elements in the **Y** matrix is shown in Fig. 4.13. There are 235683 non-zero elements.

Equation (4-27) is solved using LU factorization. However, directly applying LU factorization to the Y matrix shown in Fig. 4.13 results an enormous number (28358730+29464704-235683=57587751) of fill-in elements as shown in Fig. 4.14.

While using the LU factorization shown in Fig. 4.14 to solve (4-27), 80% of the total simulation time is consumed by the network equation solving process.

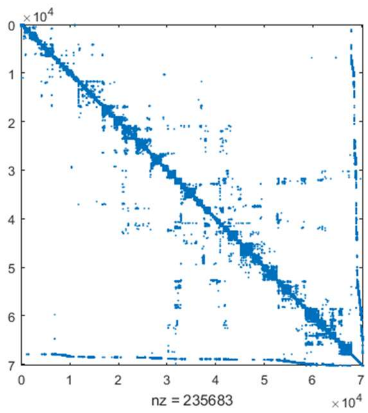
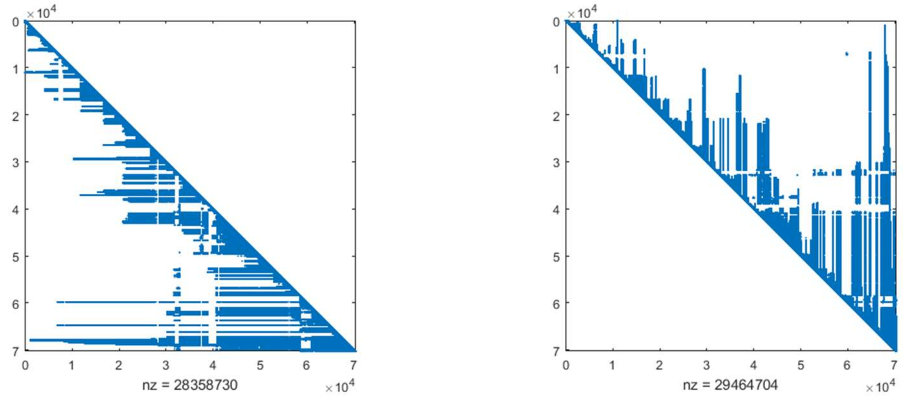


Fig. 4.13. Non-zero elements in the EI system **Y** matrix.



a) L matrix.

b) U matrix.

Fig. 4.14. LU factorization of EI system **Y** matrix.

To overcome this issue, apply a pre-conditioner called column approximate minimum degree permutation to **Y** matrix before applying LU factorization to it. The bus numbers in the **Y** matrix is reordered after that and the reordered **Y** matrix is shown in Fig. 4.15.

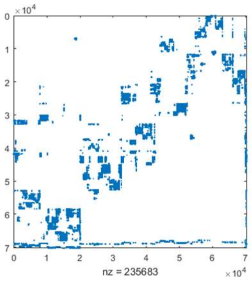
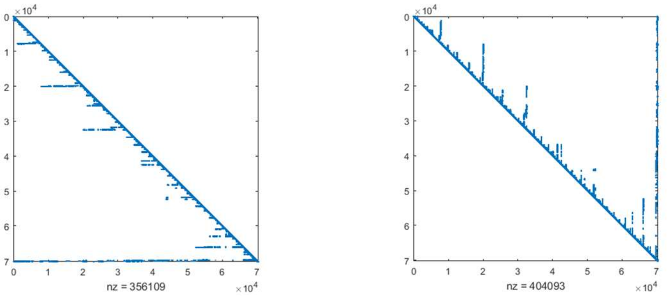


Fig. 4.15. Non-zero elements in the reordered EI system **Y** matrix.

The resulting LU factorization is shown in Fig. 4.16. The number of fill-in elements is more than 100 times smaller than the original case. (356109+404093-235683=524519)

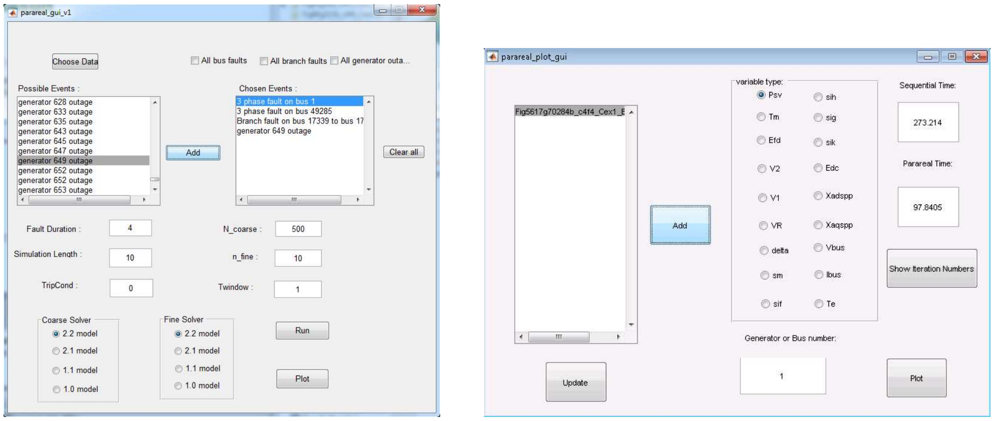


a) L matrix. b) U matrix.

Fig. 4.16. LU factorization of the reordered EI system **Y** matrix.

4.6.3 Test Results

Fig. 4.17 shows the GUI for Parareal Algorithm testing. Contingencies and be selected in the GUI shown in Fig. 4.17 a) and the simulation results can be viewed in the GUI shown in Fig. 4.17 b).



parareal_gui_v1

□ All bus faults □ All branch faults □ All generator outa...

Possible Events :

generator 628 outage
generator 633 outage
generator 635 outage
generator 643 outage
generator 645 outage
generator 647 outage
generator 649 outage
generator 652 outage
generator 652 outage
generator 653 outage

Chosen Events :

3 phase fault on bus 1
3 phase fault on bus 49285
Branch fault on bus 17339 to bus 17
generator 649 outage

Fault Duration : 4

Simulation Length : 10

TripCond : 0

N_coarse : 500

n_fine : 10

Twindow : 1

Coarse Solver: ● 2.2 model ○ 2.1 model ○ 1.1 model ○ 1.0 model

Fine Solver: ● 2.2 model ○ 2.1 model ○ 1.1 model ○ 1.0 model

parareal_plot_gui

Fig5617g70284b_c4f4_Cex1_E

variable type: ● Psv ○ Tm ○ Efd ○ V2 ○ V1 ○ VR ○ delta ○ sm ○ sif ○ sih ○ sig ○ sik ○ Edc ○ Xadspp ○ Xaqspp ○ Vbus ○ Ibus ○ Te

Sequential Time: 273.214

Parareal Time: 97.8405

Generator or Bus number: 1

a) Contingency Selection GUI.

b)Simulation results viewing GUI.

Fig. 4.17. GUI for Parareal Algorithm.

The test environment is a work station with Intel Xeon CPU E5-2650, 64 GB RAM, 16 physical cores. During the simulation, there are 16 workers in MATLAB parallel pool. It takes 273 s to simulate the first 1 s of the EI system for sequential simulation and 98 s for Parareal simulation. The iteration numbers of the different 1 s windows are given in Fig. 4.18. It shows that the theoretical speedup is 50/6 (approximately 8 times). Compare to the theoretical speedup, the actual speedup is 273/98 (approximately 2.7 times).

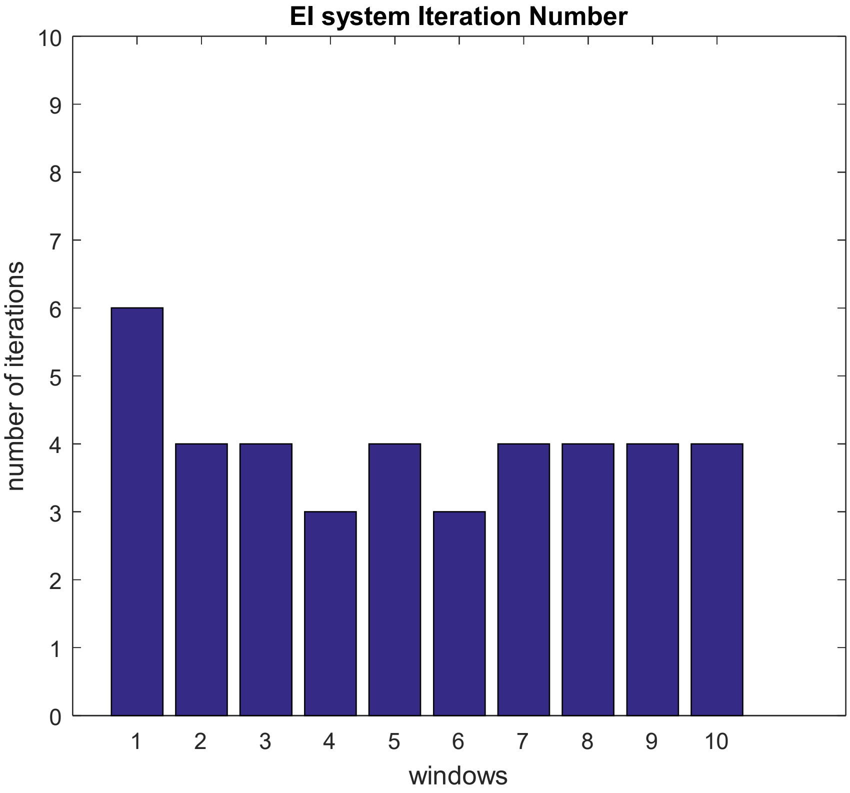


Fig. 4.18. Iteration numbers of the Parareal algorithm.

To close the gap between the actual speedup and the theoretical speedup, the communication overhead between processors has to be investigated.

The selected simulation results are shown in Fig. 4.19. The converged coarse solution agrees with the sequential simulation results.

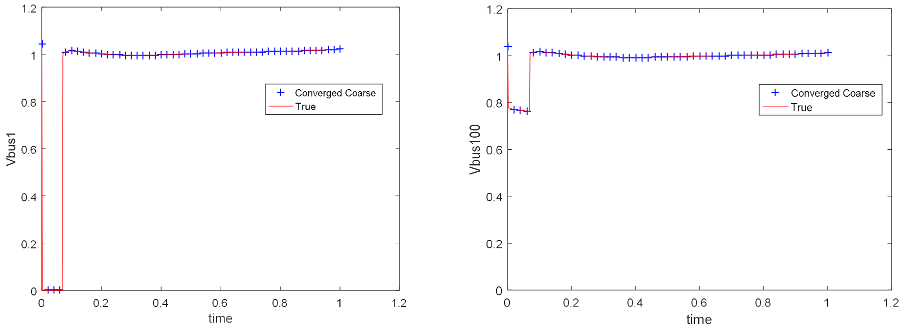


Fig. 4.19. Selected simulation results of the fault bus and a neighboring bus voltage.

4.7 Improving the Convergence Rate of Parareal-in-time Power System Simulation using the Krylov Subspace

Parareal algorithm thrives on how fast it converges. As reported in [14], if the number of iterations meets the number of coarse intervals, the theoretical speedup of parareal would downgrade to be the same as a sequential approach. Moreover, the time cost of coarse solution is neglected in theoretical time cost analysis. In fact, a large percentage of the total time could be spent on calculating a coarse solution. Some existing literature exploits the Hamiltonian nature of a system to design a Hamiltonian projection enhanced parareal to improve its convergence rate [63]. However, since power systems are not Hamiltonian, such type of enhancement is not applicable. Ad hoc solution of parareal is needed for fully realizing the potential of parareal in power system simulation applications.

Building a Krylov subspace using the coarse solution from the past iterations has been reported in [64]-[66] as an effective tool to speed up the convergence of parareal algorithm. The convergence accelerator improves the accuracy of the coarse solution to the level of fine solution by replacing the coarse propagation with the propagation of the basis of Krylov subspace using fine solver, which can be computed in parallel. In this dissertation, the plain parareal approach is augmented with a Krylov subspace based convergence accelerator to simultaneously improve the convergence rate and reduce the coarse time cost for power system simulation. Instead of abandoning the coarse solutions from previous iterations, the proposed approach saves them to span a Krylov subspace. Then, the basis of the Krylov subspace is propagated by using the fine solver to create a new space to which the coarse solution of the latest iteration can be projected. The proposed approach improves the accuracy of the coarse propagation to the level of fine propagation and, therefore, reduces the number of iterations. The coarse solver is only used once for the initial guess. The subsequent coarse propagations are replaced by fine propagations of the Krylov subspace's basis. Those fine propagations are independent tasks that can be assigned to parallel processors.

4.7.1 Projection to the Krylov Subspace

There are two subspaces. One is built from the coarse solutions but, in practice, another subspace built on the fine solution (the last point of each fine trajectory, to be precise) turns out to be more useful for nonlinear system case.

Denote the space spanned from coarse solutions as,

$$\mathcal{S}^k = \text{span}\left\{\mathbf{q}_i^j : i = 0,\ldots,N_c - 1, j = 0,\ldots,k\right\} \tag{4-28}$$

Denote the space spanned from fine solutions as,

$$\mathcal{F}(\mathcal{S}^k) = \text{span}\left\{\tilde{\mathbf{q}}_i^j : i = 0,\ldots,N_c - 1, j = 0,\ldots,k\right\} \tag{4-29}$$

In the Krylov enhanced coarse correction step,

$$\mathcal{K}_{\Delta t}(\mathbf{q}, t_{i+1}, t_i) := \mathcal{G}_{\Delta t}((\mathbf{I} - \mathbf{P}^k)\mathbf{q}, t_{i+1}, t_i) + \mathcal{F}_{\delta t}(\mathbf{P}^k\mathbf{q}, t_{i+1}, t_i) \tag{4-30}$$

The basis of $\mathcal{S}^k$, $\mathbf{s}_1,\ldots,\mathbf{s}_r$ will be used to calculate $\mathbf{P}^k$,

$$\mathbf{P}^k = \mathbf{S}_k(\mathbf{S}_k^T\mathbf{S}_k)^{-1}\mathbf{S}_k^T \tag{4-31}$$

where $\mathbf{S}_k = [\mathbf{s}_1 \quad \mathbf{s}_2 \quad \ldots \quad \mathbf{s}_r]$.

The basis of $\mathcal{F}(\mathcal{S}^k)$, $\tilde{\mathbf{s}}_1,\ldots,\tilde{\mathbf{s}}_r$ will be used to bypass $\mathcal{F}_{\delta t}(\mathbf{P}^k\mathbf{q}, t_{i+1}, t_i)$:

$$\mathcal{F}_{\delta t}(\mathbf{P}^k\mathbf{q}, t_{i+1}, t_i) = \mathcal{F}_{\delta t}(\sum_{i=1}^{r}\alpha_i s_i, t_{i+1}, t_i) = \sum_{i=1}^{r}\alpha_i\mathcal{F}_{\delta t}(s_i, t_{i+1}, t_i) \tag{4-32}$$

Note that $\mathbf{P}^k\mathbf{q} \in \mathcal{S}^k$, therefore,

$$\mathcal{F}_{\delta t}(\sum_{i=1}^{r}\alpha_i s_i, t_{i+1}, t_i) \in \mathcal{F}(\mathcal{S}^k) \tag{4-33}$$

where

$$\sum_{i=1}^{r}\alpha_i\mathcal{F}_{\delta t}(s_i, t_{i+1}, t_i) = \sum_{i=1}^{r}\alpha_i\tilde{s}_i \tag{4-34}$$

In practice, α_i does not need to be calculated explicitly. The purpose of defining α_i is to illustrate (4-33).

The additional computation is equivalent to applying fine solver $\mathcal{F}_{\delta t}(\cdot)$ to r more coarse intervals (r is the column rank of basis matrix $\mathbf{S}^k$). Since the initial values of those propagations are the elements $\mathbf{s}_1,\ldots,\mathbf{s}_r$ of the basis $\mathbf{S}^k$, those computations can be executed in parallel during fine propagations.

The new correction step with Krylov subspace based convergence accelerator is,

$$\mathbf{q}_{i+1}^{k+1}=\mathcal{K}_{\Delta t}(\mathbf{q}_i^{k+1},t_{i+1},t_i)-\tilde{\mathbf{q}}_{i+1}^{k}+\mathcal{K}_{\Delta t}(\mathbf{q}_i^{k},t_{i+1},t_i) \tag{4-35}$$

Since power systems are inhomogeneous, one more step is required to compensate the inhomogeneous input to the system (i.e. control reference input of power control devices)

$$\mathcal{G}_{\Delta t}(\mathbf{0},t_{i+1},t_i) \tag{4-36}$$

The aforementioned procedure is designed for the linear system. Since for nonlinear system,

$$\sum_{i=1}^{r}\alpha_i\mathcal{F}_{\delta t}(s_i,t_{i+1},t_i)\neq\sum_{i=1}^{r}\alpha_i\tilde{s}_i \tag{4-37}$$

the practical convergence rate improvement lies between the linear system speedup and the plain parareal. In terms of computational structure, the Krylov subspace convergence accelerator requires N_{b} = dim($\mathbf{S}^k$) more worker processors to propagate the basis in addition to the Nc worker processors for the plain parareal algorithm.

4.7.2 Construction of Krylov Subspace and Its Basis

The Krylov subspace is built by adding new vectors from new iterations of the coarse solution. Therefore, after the k-th iteration, the Krylov subspace is represented by matrix $\mathbf{K}^k$

$$\mathbf{K}^k = \begin{bmatrix} \mathbf{q}_0^0 & \cdots & \mathbf{q}_{N_c-1}^0 & \cdots & \mathbf{q}_0^k & \cdots & \mathbf{q}_{N_c-1}^k \end{bmatrix} \tag{4-38}$$

The basis of $\mathbf{K}^k$ can be calculated through singular value decomposition (SVD),

$$\mathbf{K}^k = \mathbf{U\Sigma V}^T \tag{4-39}$$

If the rank of $\mathbf{U}$ is r, then the first r columns of $\mathbf{U}$ form the basis of the vector space $\mathbf{K}^k$. The computation of SVD can be done by one of many fast large-scale SVD algorithms, such as the bi-diagonalization algorithm proposed in [67].

4.7.3 Computational Structure

The proposed convergence accelerator requires marginal computational resources in the master processor.The calculation of the basis is performed by fast SVD algorithms, as mentioned above. The proposed approach replaces the time consuming and sequential coarse propagation with a much cheaper coarse projection step, which improves the speed of every iteration significantly. The majority of the additional computation of propagating the basis of the Krylov subspace is performed by additional worker processors. In the case of modern HPC where the number of available worker processors is no longer a limiting factor, an increase in worker processors is an acceptable price to pay for the improvement of the coarse solution accuracy. A diagram of the parareal algorithm with the convergence accelerator (highlighted in red) and without it is presented in Fig. 4.20.

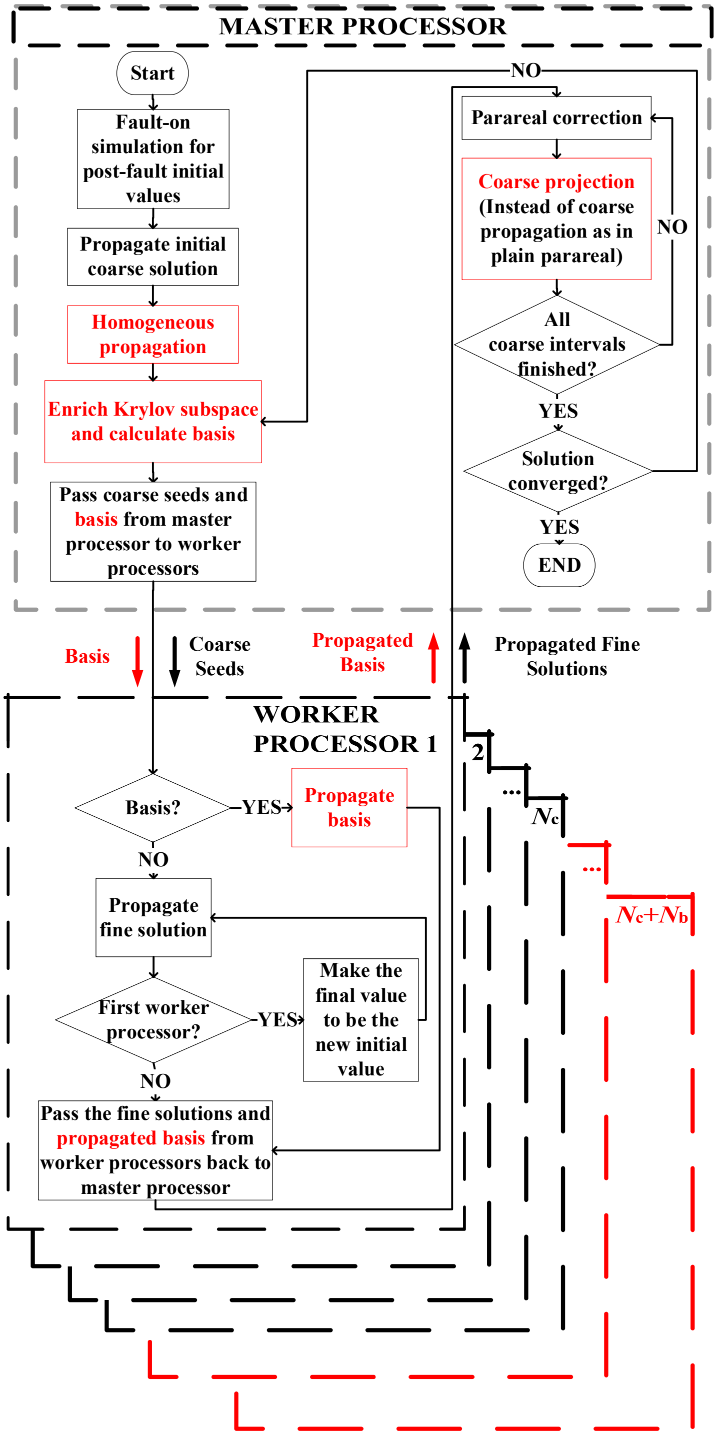

MASTER PROCESSOR
Start
Fault-on simulation for post-fault initial values
Propagate initial coarse solution
Homogeneous propagation
Enrich Krylov subspace and calculate basis
Pass coarse seeds and basis from master processor to worker processors
NO
Parareal correction
Coarse projection (Instead of coarse propagation as in plain parareal)
NO
All coarse intervals finished?
YES
Solution converged?
YES
END
Basis
Coarse Seeds
Propagated Basis
Propagated Fine Solutions
WORKER PROCESSOR 1
2
...
Nc
...
Nc+Nb
Basis?
YES
Propagate basis
NO
Propagate fine solution
First worker processor?
YES
Make the final value to be the new initial value
NO
Pass the fine solutions and propagated basis from worker processors back to master processor

Fig. 4.20. Computational structure of parareal algorithm with convergence accelerator.

4.7.4 Relation between the Growth of the Krylov Subspace and Convergence Rate

The Krylov subspace-enhanced parareal is applied to the SMIB in (4-40), deliberately choosing a large coarse step size of 0.5s to reduce the accuracy of the coarse solver. The converged simulation result of rotor angle δ is shown in Fig. 4.21. The plain parareal converges in 7 iterations and the parareal with Krylov subspace projection converges in 4 iterations.

$$\begin{cases} \dot{\delta} = \Delta\omega \\ 2H\dot{\omega} = P - D\Delta\omega - EB\sin\delta - EG\cos\delta \end{cases} \tag{4-40}$$

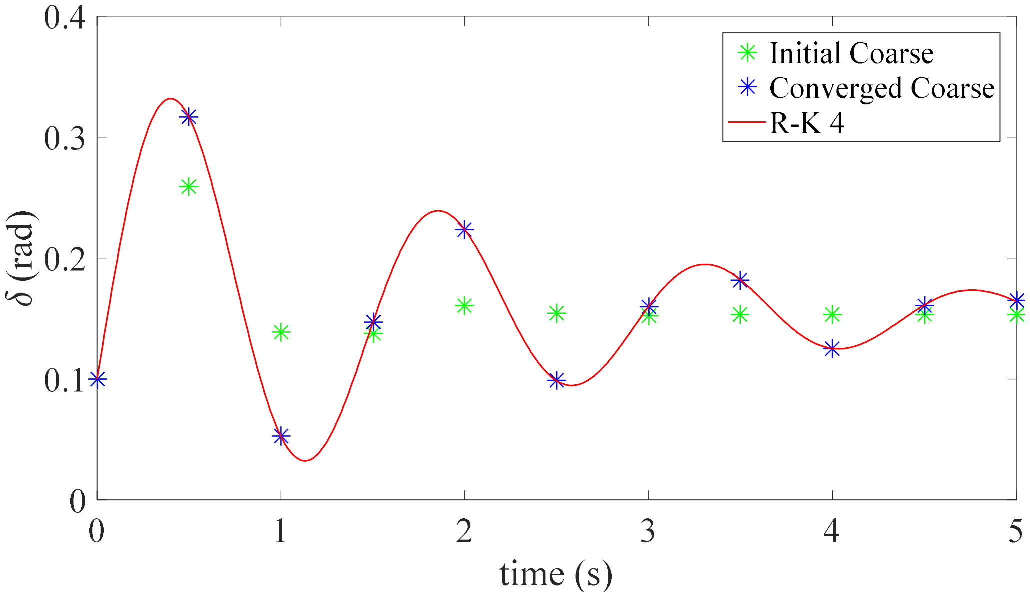


Fig. 4.21.Rotor angle simulation result from the Krylov subspace enhanced parareal approach.

The basis of the Krylov subspace shows similar convergence behavior. As shown in Fig. 4.22, the basis converges close to unit vectors $[0\ 1]^T$ and $[1\ 0]^T$ after 4 iterations. Fig. 4.23 shows that the error decreases much slower for the plain parareal algorithm.

The reason is that the coarse propagation step of the plain parareal has a lower accuracy than the coarse projection step of the proposed approach. The largest change of the basis occurs between iteration 1 and iteration 2, which is consistent with the decrease of the coarse solution error as shown in Fig. 4.23.

In terms of numerical accuracy, undamped system is more challenging to simulate, because the numerical error is maintained and accumulated without being reduced by the system damping. Moreover, the idea of Krylov subspace is proposed under linear system assumption and its performance may downgrade under large disturbance.

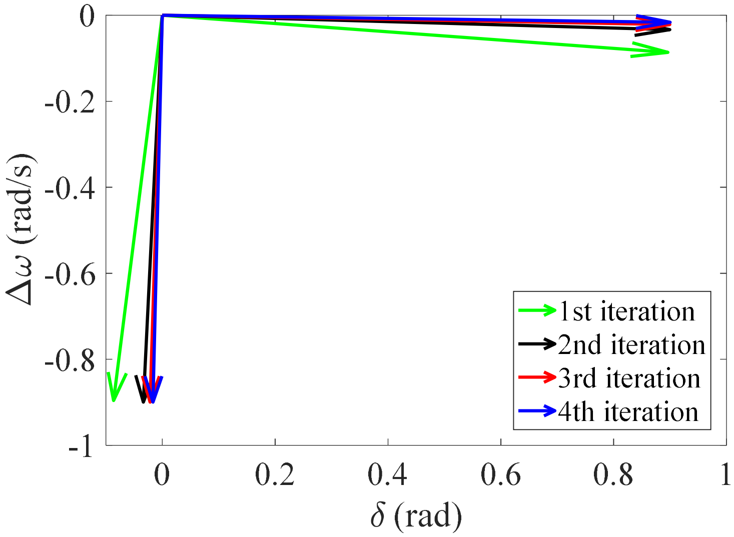


Fig. 4.22. Krylov subspace’s basis for different iterations of the damped SMIB system simulation.

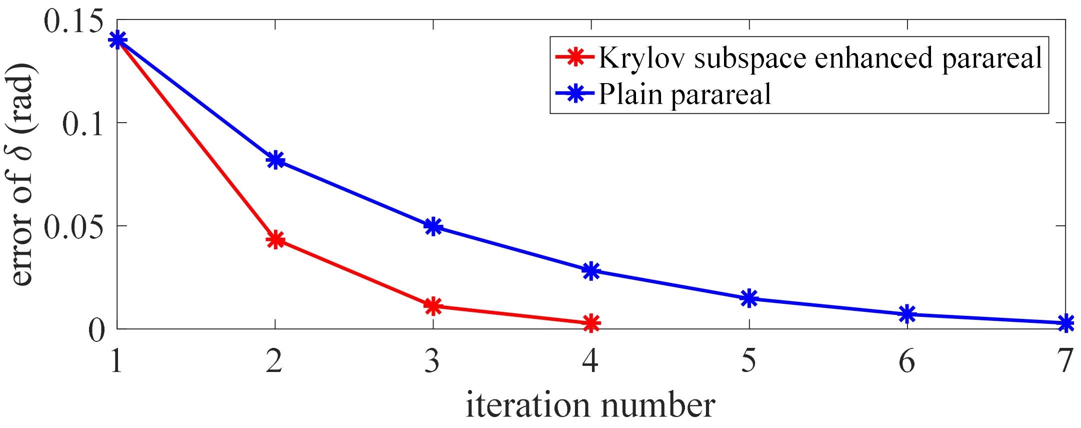


Fig. 4.23. Error comparison between the Krylov subspace-enhanced parareal and the plain parareal for each iteration (damped SMIB system).

To evaluate how the proposed convergence accelerator performs under such conditions, the *D* parameter in (4-40) is set to zero and a large disturbance to induce marginally stable oscillation is tested. As expected, both the plain parareal and the proposed approach takes more iterations to converge. For this extreme case, the plain parareal outperforms the proposed approach. However, the proposed approach still provides accurate system response. The simulation results for rotor angle δ are shown in Fig. 4.24 and the numbers of iterations required are listed in Table 4.6.

Table 4.6. Numbers of iterations.

Disturbance	Plain Parareal	Krylov Subspace enhanced Parareal
Small	7	4
Large	8	10

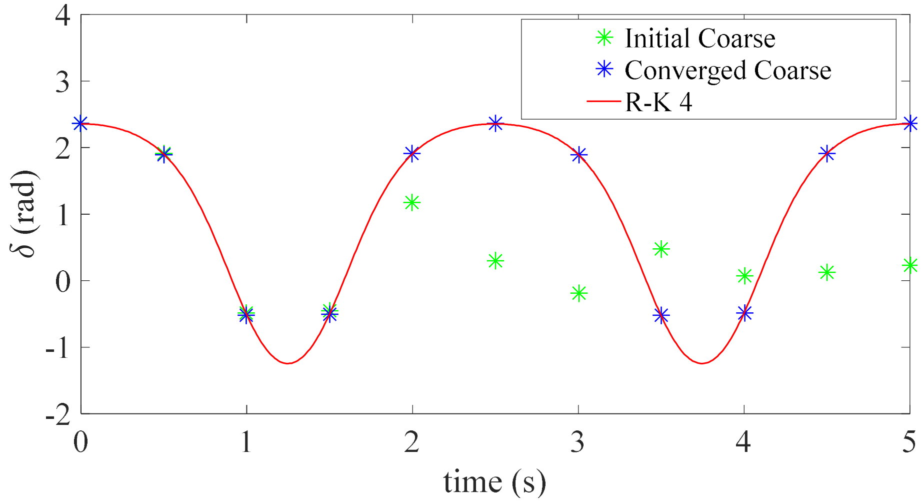


Fig. 4.24. Rotor angle simulation result from the Krylov subspace-enhanced parareal approach for the undamped system under large disturbance

In contingency screening applications, one more procedure can be added to help making the decision of whether the plain parareal or the proposed approach should be applied to a specific contingency as shown in Fig. 4.25. This step checks the initial coarse propagation. If its rotor angle is larger than a certain threshold, then the plain parareal should be used; otherwise, the proposed approach should be used. Since there is only a small number of such extreme cases in real life operation studies, the proposed approach can still benefit the simulation of the majority of contingencies.

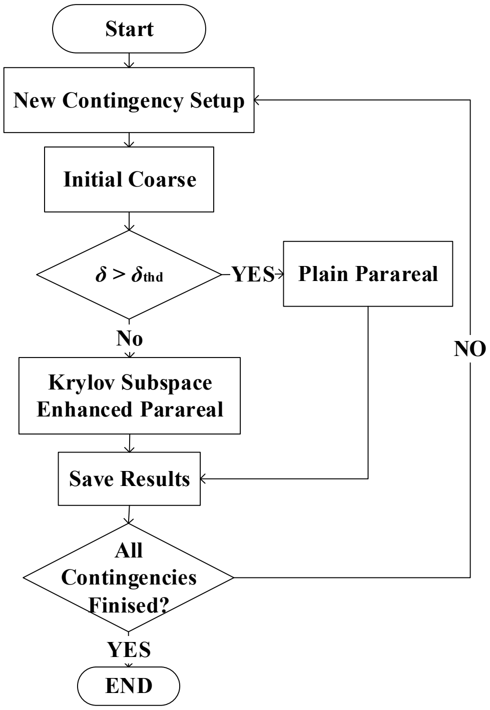


Fig. 4.25. Contingency screening procedure having the options of both plain parareal and the Krylov subspace-enhanced parareal.

4.7.5 Case Study

To show the generality of the proposed approach, it is tested on the IEEE 10-machine 39-bus system. All generators are represented by the classical model. A 3-phase bus fault is applied at bus 1 and cleared after 10 cycles. The fine step size is 0.01 s and the coarse step size is 0.1 s. The simulation result for the rotor speed deviations is shown in Fig. 4.26.

The error comparison of both the plain parareal and the proposed approach is shown in Fig. 4.27. The proposed approach only takes 4 iterations compared to 8 iterations required for the plain parareal to achieve the same level of accuracy.

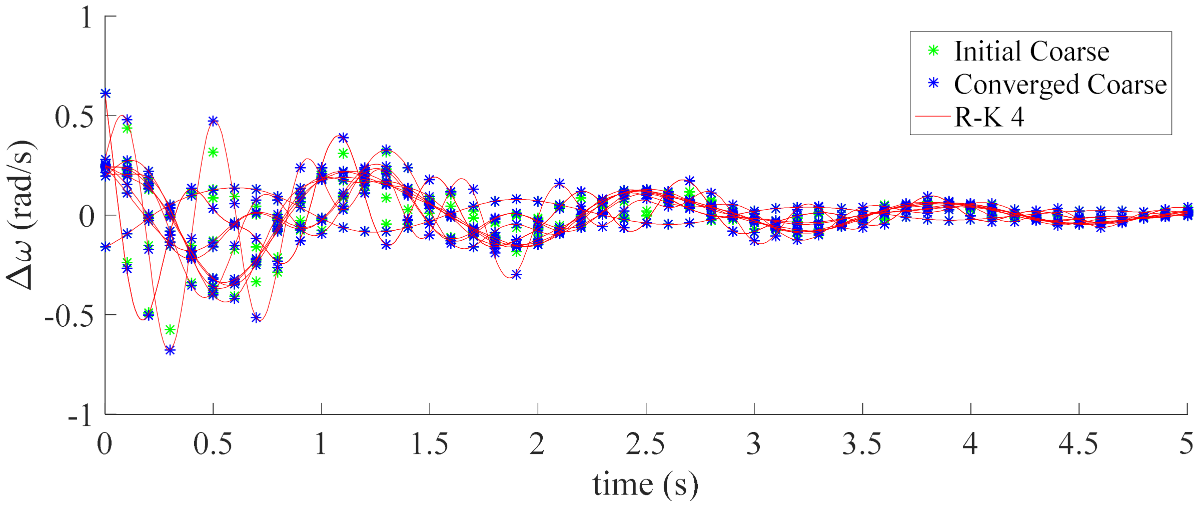


Fig. 4.26. IEEE 10-machine 39-bus system rotor speed deviation simulation results from the Krylov subspace-enhanced parareal approach.

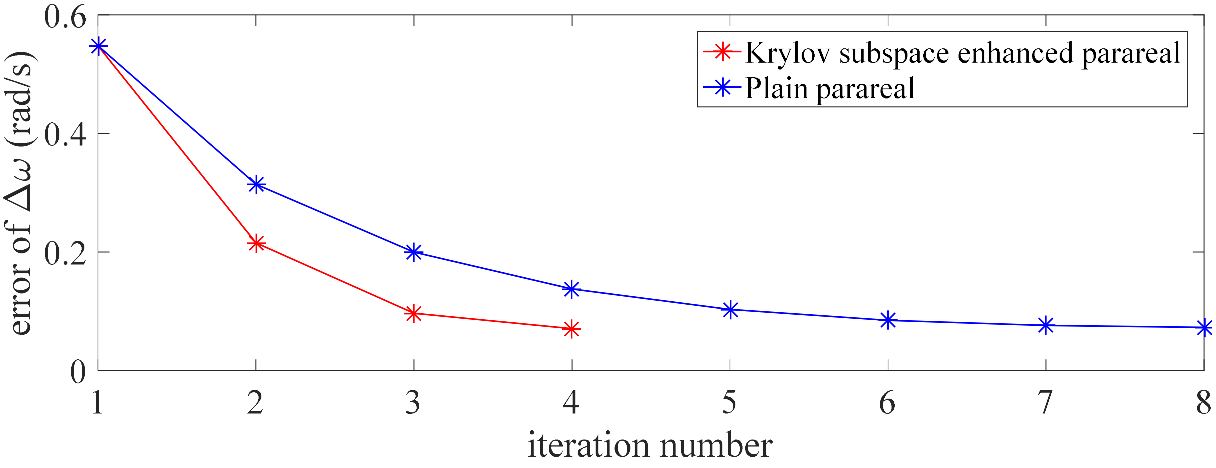


Fig. 4.27. Error comparison between the Krylov subspace-enhanced parareal and the plain parareal for each iteration (IEEE 10-machine 39-bus system).

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORK

This dissertation investigates the feasibilities of two approaches for achieving faster-than-real-time power system simulation. The first one is a semi-analytical approach, by which an SAS is derived for power system DAEs having the form of a summation of many CUs to be computed in parallel by parallel processors. The second approach is temporal decomposition of the simulation period using the Parareal-in-time algorithm. For both approaches, the time performances demonstrated in MATLAB environment suggest promising speedup of power system simulation.

To further reduce the gap to faster-than-real-time power system simulation, both approaches will need to be implemented on high-performance computers using professional languages for parallel computing, and the communications among parallel processors will need to be optimized to reduce the computation overhead.

LIST OF REFERENCES

[1] W. Ju, K. Sun, J. Qi, "Multi-Layer Interaction Graph for Analysis and Mitigation of Cascading Outages," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 7, No. 2, pp. 239-249, Jun. 2017.

[2] J. Qi, W. Ju, K. Sun, "Estimating the Propagation of Interdependent Cascading Outages with Multi-Type Branching Processes," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 1212-1223, Mar. 2017.

[3] J. Qi, W. Huang, K. Sun, W. Kang, "Optimal Placement of Dynamic Var Sources by Using Empirical Controllability Covariance," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 240-249, Jan. 2017.

[4] W. Huang, K. Sun, J. Qi, J. Ning, "Optimal Allocation of Dynamic Var Sources Using the Voronoi Diagram Method Integrating Linear Programing," *IEEE Trans. Power Syst.*, vol. 32, No. 6, pp. 4644 – 4655, Nov. 2017

[5] B. Wang, K. Sun, "Formulation and Characterization of Power System Electromechanical Oscillations," *IEEE Trans. Power Syst.*, vol. 61, no. 6, pp. 5082-5093, Nov. 2016.

[6] B. Wang, X. Su, K. Sun, "Properties of the Frequency-Amplitude Curve," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 826-827, Jan. 2017.

[7] S. You, J. Guo, W.Yao, S. Wang, Y. Liu, Y.Liu, "Ring-down oscillation mode identification using multivariate Empirical Mode Decomposition," IEEE PES General Meeting, Boston, MA, 2016.

[8] M. Wazwaz, "A comparison between the variational iteration method and Adomian decomposition method," *Journal of Computational and Applied Mathematics*, vol. 207, no. 1, pp. 129-136, 2007.

[9] Y. Tan, A. Saeid, "Homotopy analysis method for quadratic Riccati differential equation," Communications in Nonlinear Science and Numerical Simulation, vol. 13, no. 3, pp. 539-546, 2008.

[10] K. Abbaoui, Y. Cherruault, "New ideas for proving convergence of decomposition methods," Computers & Mathematics with Applications, vol. 29, no. 7, pp. 103-108, 1995.

[11] K. Abbaoui, Y. Cherruault, "Convergence of Adomian's method applied to differential equations," Computers & Mathematics with Applications, vol. 28, no. 5, pp. 103-109, 1994.

[12] A. S. Mahmood, L. Casasús, W. Al-Hayani, "The decomposition method for stiff systems of ordinary differential equations," Applied mathematics and computation, vol. 167, no. 2, pp. 964-975, 2005.

[13] J. Li, "Adomian's decomposition method and homotopy perturbation method in solving nonlinear equations," Journal of Computational and Applied Mathematics, vol. 228, no. 1, pp. 168-173, 2009.

[14] J. Lions, Y. Maday, and G. Turinici, "A parareal in time discretization of pde's," C. R. Acad Sci. Paris, Series I, vol. 332, pp. 661–668, 2001.

[15] C. Farhat, J. Cortial, C. Dastillung, and H. Bavestrello, "Time-parallel implicit integrators for the near-real-time prediction of linear structural dynamic

responses," *International Journal for Numerical Methods in Engineering*, vol. 67, no. 5, pp. 697–724, 2006.

[16] M. L. Crow, M. Ilic, "The parallel implementation of the waveform relaxation method for transient stability simulations, " IEEE Trans. Power Syst., vol. 5, no. 3, pp. 922-932, Aug. 1990.

[17] J. Qi, J. Wang, H. Liu, and A. D. Dimitrovski, "Nonlinear model reduction in power systems by balancing of empirical controllability and observability covariances," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 114–126, Jan. 2017.

[18] H. Jafari, C.M. Khalique and M.Nazari, "Application of the Laplace decomposition method for solving linear and nonlinear fractional diffusion-wave equations," *Applied Mathematics Letter*, vol. 24, pp. 1799–1805, Jan. 2011.

[19] S. A. Khuri, "A Laplace decomposition algorithm applied to a class of nonlinear differential equations," *J. Applied Mathematics*, vol. 1, pp. 141–155, Jun. 2001.

[20] J. Biazar and S.M. Shafiof, "A simple algorithm for calculating Adomian polynomials," *Int. J. Contemp. Math. Sciences*, vol. 2, no. 20, pp. 975–982, 2007.

[21] A. Wazwaz, "The modified decomposition method and Pade approximants for solving the Thomas Fermi equation," *Applied Math. and Computation*, vol. 105, no. 1, pp. 11–19, Jun. 1999.

[22] J. Qi, K. Sun and W. Kang, "Optimal PMU Placement for Power System Dynamic State Estimation by Using Empirical Observability Gramian," *IEEE Trans. Power Syst.*, vol.30, pp.2041-2054, July 2015.

[23] J. H. Chow and K. W. Cheung, “A toolbox for power system dynamics and control engineering education and research,” *IEEE Trans. Power Syst.*, vol. 7, no. 4, pp. 1559–1564, Nov. 1992.

[24] G. Adomian, "On the convergence region for decomposition solutions,"*J. Comput. & Applied Math.*, vol. 11, pp. 379–380, 1984.

[25] J.S. Duan, et al, "New higher-order numerical one-step methods based on the Adomian, and the modified decomposition methods," *Applied Math. and Computation*, vol. 218, pp. 2810–2828, 2011.

[26] S. M. Holmquist, “An examination of the effectiveness of the Adomian decomposition method in fluid dynamic applications,” Ph.D. dissertation, Dept. Math., Univ. of Central Florida, Orlando, FL, 2007.

[27] L. A. Bougoffa, R.C. Rach and S. El-manouni, "A convergence analysis of the Adomian decomposition method for an abstract Cauchy problem of a system of firstorder nonlinear differential equations," *Int. J. Comput. Mathematics*, vol. 2, no. 90, pp. 360–375, Aug. 2012.

[28] N. Duan and K. Sun, “Application of the Adomian Decomposition Method for Semi-Analytic Solutions of Power System Differential Algebraic Equations,” *Powertech*, Eindhoven, 2015.

[29] O.T. Kolebaje, O.L. Akinyemi and R.A. Adenodi, "On the application of the multistage laplace adomian decomposition method with pade approximation to the rabinovich-fabrikant system," *Advances in Applied Science Research*, vol. 4, no. 3, pp. 232–243, 2013.

[30] J. S. Duan, R. Rach, et al., "A review of the Adomian decomposition method and its applications to fractional differential equations," *Commun. Frac. Calc.*, vol. 3, no. 2, pp. 73–99, 2012.

[31] J. S. Duan, R. Rach and A.M. Wazwaz, "A reliable algorithm for positive solutions of nonlinear boundary value problems by the multistage Adomian decomposition method," *Applied Mathematical Modelling*, vol. 37, no. 20, pp. 8687–8708, Jan.

[32] P. M. Anderson and A.A. Fouad, *Power system control and stability.* 2nd ed., New York: Wiley Interscience, 2003, pp.13-66.

[33] J. I. Ramos, "Piecewise-adaptive decomposition methods," *Chaos, Solitons & Fractals*, vol. 40, no. 4, pp. 1623-1636, 2009.

[34] M. Kezunovic, "Monitoring of power system topology in real-time," in *Proc. 39th Hawaii Int. Conf. System Sciences*, Jan. 2006.

[35] I. Hiskens, J. Alseddiqui, "Sensitivity, approximation and uncertainty in power system dynamic simulation," IEEE Trans. Power Systems, vol. 21, no. 4, pp. 1808-1820, Nov. 2006.

[36] M. Tatari, M. Dehghan, M. Razzaghi, "Application of the Adomian decomposition method for the Fokker-Planck equation," Math. and Comput. Modelling, vol. 45, no. 5, pp.638-650, Mar. 2007.

[37] B. Spencer, Jr., L. Bergman, "On the numerical solution of the Fokker-Planck equation for nonlinear stochastic systems," Nonlinear Dynamics, vol. 4, no. 4, pp. 357-372, Aug. 1993.

[38] Y. Saito, T. Mitsui, "Simulation of stochastic differential equations," Ann. Inst. Stat. Math., vol. 45, no. 3, pp. 419-432, Sep. 1993.

[39] D. J. Higham, "An algorithmic introduction to numerical simulation of stochastic differential equations," Soc. Ind. and Appl. Math. Review, vol.43, no.3, pp.525-546, 2001.

[40] K. Wang, M. Crow, "Fokker Planck equation application to analysis of a simplified wind turbine model," North American Power Symposium, Champaign, IL, 2012.

[41] F. Milano, R. Zarate-Minano, "A systematic method to model power systems as stochastic differential algebraic equations," IEEE Trans. Power Systems, vol. 28, no. 4, pp. 4537-4544, Nov. 2013.

[42] W. Wu, K. Wang, G. Li, Y. Hu, "A stochastic model for power system transient stability with wind power," IEEE PES General Meeting, National Harbor, MD, 2014.

[43] K. Wang, M. Crow, "Numerical simulation of stochastic differential algebraic equations for power system transient stability with random loads," IEEE PES General Meeting, Detroit, MI, 2011.

[44] B. Yuan, M. Zhou, G. Li, X. Zhang, "Stochastic small signal stability of power system with wind power generation," IEEE Trans. Power Systems, vol. 30, no. 4, pp.1680-1689, Jul. 2015.

[45] X. Wang, H. D. Chiang, J. Wang, H. Liu, T. Wang, "Long-term stability analysis of power systems with wind power based on stochastic differential equations: model

development and foundations," IEEE Trans. Sustainable Energy, vol. 6, no. 4, pp. 1534-1542, Oct. 2015.

[46] N. Duan, K. Sun, “Power system simulation using the multi-stage Adomian decomposition method,” *IEEE Trans. Power Systems*, vol. 32, no. 1, pp. 430-441, January 2017

[47] G. Adomian, "Nonlinear stochastic differential equations," Math. Anal. And Applicat., vol. 55, no. 2, pp. 441-452, Aug. 1976.

[48] F. Galiana, E. Handschin, A. Fiechter, "Identification of stochastic electric load models from physical data," IEEE Trans. Automat. Control, vol. 19, no. 6, pp. 887-893, Dec. 1974.

[49] T. Sauer, "Numerical solution of stochastic differential equations in finance," in Handbook of Mathematical Functions, Springer Berlin Heidelberg, 2012, pp. 529-550.

[50] K. Nouri, "Study on stochastic differential equations via modified Adomian decomposition method," U. P. B. Sci. Bull., Series A, vol. 78, no. 1, pp. 81-90, 2016.

[51] W. Cao, M. Liu, Z. Fan, "MS-stability of the Euler-Maruyama method for stochastic differential delay equations," Appl. Math. and Comp., vol. 159, no. 1, pp. 127-135, 2004.

[52] M. Hutzenthaler, A. Jentzen, P. Kloeden, "Strong convergence of an explicit numerical method for SDEs with non-globally Lipschitz continuous coefficients", The Ann. of Appl. Probability, vol. 22, no.4, pp. 1611-1641, 2012.

[53] G. Adomian, "A review of the decomposition method in applied mathematics," J. of Math. Anal. and Applicat., vol. 135, no.2, pp. 501-544, Nov. 1988.

[54] U. H. Thygesen, "A survey of Lyapunov techniques for stochastic differential equations," Dept. of Math. Modelling, Tech. Univ. of Denmark, Lyngby, Denmark, Rep. IMM Technical Report nr. 18-1997.

[55] X. Mao, Stochastic Differential Equations and Applications, 2nd edition, Chichester, UK, Horwood, 2008.

[56] K. Burrage, P. Burrage, T. Mitsui, "Numerical solutions of stochastic differential equations – implementation and stability issues," J. of Computational and Appl. Math., vol. 125, no. 1, pp. 171-182, 2000.

[57] F. Kozin, "A survey of stability of stochastic systems," Automatica, vol. 5, no. 1, pp. 95-112, Jan. 1969.

[58] A. S. Nielsen, “Feasibility study of the parareal algorithm,” Master’s thesis, Tech. Univ. Denmark, Kongens Lyngby, Denmark, 2012.

[59] G. Gurrala, A. Dimitrovski, P. Sreekanth, S. Simunovic and M. Starke, "Parareal in Time for Fast Power System Dynamic Simulations," *IEEE Trans. Power System*, vol. 99, pp. 1-11, Jul. 2015.

[60] K. R. Padiyar, Power System Dynamics Stability and Control. Hyderabad, India: B.S. Publications, 2002.

[61] K. Shubhanga and B. Papa Rao, Manual for A Multi-machine Transient Stability Programme, PART-I Symmetrical Fault, Dept. Elect. Eng., NITK. Surathkal, India [Online]. https://www.ee.iitb.ac. in/anil/download/Transient_Stability_Programs/

[62] J. Qi, J. Wang, H. Liu, and A. D. Dimitrovski, “Nonlinear model reduction in power systems by balancing of empirical controllability and observability covariances,” *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 114–126, Jan. 2017.

[63] X. Dai, C. Le Bris, F. Legoll, Y. Maday, “Symmetric parareal algorithms for Hamiltonian systems,” *Math. Modelling and Numerical. Anal.*, vol. 47, no.3, pp. 717-742, 2013.

[64] C. Farhat, J. Cortial, C. Dastillung, H. Bavestrello, “Time-parallel implicit integrators for the near-real-time prediction of linear structural dynamic responses,” *Int. J. for Numerical Methods in Eng.*, vol. 67, no. 5, pp. 697–724, 2006.

[65] M. Gander, M. Petcu, “Analysis of a Krylov subspace enhanced parareal algorithm for linear problems,” *Math. Modelling and Numerical. Anal.*, vol. 25, pp. 114-129, 2008.

[66] F. Chen, J. S. Hesthaven, X. Zhu, “On the use of reduced basis methods to accelerate and stabilitze the parareal method,” in *Reduced Order Methods for Modeling and Computational Reduction*, Springer International Publishing, 2014, pp. 187-214.

[67] N. Bosner, “,” Ph.D. dissertation, Dept. Math., Univ. of Zagreb, Zagreb, Croatia, 2006.

APPENDIX

Publications

Patents:

[P1] **N. Duan**, C. Baone, A. Menon, M. Dokucu, Power System Disturbance Based Model Validation and Calibration, US Patent Application

[P2] **N. Duan**, K. Sun, Transient Stability Simulation and Operation of Power Systems, US Patent Application, No. 14/956,076, Dec 2015

Transaction Papers:

[J1] **N. Duan**, K Sun, "Stochastic power system simulation using the Adomian decomposition method," *IEEE Transactions on Smart Grid*, under review.

[J2] B. Wang, **N. Duan**, K Sun, "Time series based semi-analytical solution for power system simulation," *IEEE Transactions on Power Systems*, in revision.

[J3] **N. Duan**, K. Sun, "Power system simulation using the multi-stage Adomian decomposition method," *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 430-441, January 2017

Conference Papers:

[C1] **N. Duan**, W. Huang, K. Sun, X. Luo, Q. Zhang, "Multiple-shaft turbine model reduction by fractional-order modeling," submitted to 2018 IEEE PES General Meeting.

[C2] **N. Duan**, S. Simunovic, A. Dimitrovski, K. Sun, "Improving the convergence rate of parareal-in-time power system simulation using the Krylov subspace," submitted to 2018 IEEE PES General Meeting.

[C3] D. Osipov, **N. Duan**, A. Dimitrovski, S. Allu, S. Simunovic, K. Sun, "Adaptive model reduction for parareal in time method for transient stability simulations," submitted to 2018 IEEE PES General Meeting.

[C4] **N. Duan**, C. Baone, M. Dokucu, A. Menon, “A framework for synchrophasor based model calibration leveraging multiple events,” submitted to ISGT 2018, Washington, DC

[C5] **N. Duan**, A. Dimitrovski, S. Simunovic, K. Sun, J. Qi, J. Wang, “Embedding spatial decomposition in parareal in time power system simulation,” ISGT Europe 2017, Torino, Italy

[C6] **N. Duan**, A. Dimitrovski, S. Simunovic, K. Sun, “Applying reduced generator models in the coarse solver of parareal in time parallel power system simulation,” ISGT Europe 2016, Ljubljana, Slovenia

[C7] A. Teron, A. Bartlett, **N. Duan**, K. Sun, “Estimating the nonlinear oscillation frequency of a power system using the harmonic balanced method,” 2016 IEEE PES General Meeting, Boston, MA

[C8] **N. Duan**, B. Wang, K. Sun, J. Ning, “Analysis of power system oscillation frequency using differential Groebner basis and the harmonic balance method”, 2015 IEEE PES General Meeting, Denver, CO

[C9] **N. Duan**, K. Sun, “Application of the Adomian decomposition method for semi-analytic solutions of power system differential algebraic equations”, Powertech 2015, Eindhoven

VITA

Nan Duan received his B.S. in automation from Beijing University of Technology, Beijing, China and M. Eng in Control Engineering from Beijing University of Aeronautics and Astronautics, Beijing, China in 2010 and 2013 respectively. He is currently working toward his Ph.D. in the Department of Electrical Engineering and Computer Science at the University of Tennessee, Knoxville. His research interests include power system dynamic performance, power system fast time domain simulation.