DEVELOPMENT OF AN IMPLICITLY COUPLED ELECTROMECHANICAL AND ELECTROMAGNETIC TRANSIENTS SIMULATOR FOR POWER SYSTEMS

BY

SHRIRANG ABHYANKAR

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering
in the Graduate College of the
Illinois Institute of Technology

Approved ______________________
Advisor

Chicago, Illinois
December 2011

ACKNOWLEDGMENT

This page is the most important page of this document as it gives me an opportunity to thank all the people without whom this research work would have been impossible. I am grateful to all these people and many more who have contributed in their own ways technically, socially, psychologically or philosophically.

My parents, Mr. Gangadhar S. Abhyankar and Mrs. Sugandha G. Abhyankar, are the inspiration for my work. It is due to their nurturing, support and encouragement that i've reached this stage and i want to dedicate this work to them.

My wife, Archana, who has beared me through the thick and thin for all and deserves more credit than me for this work. Her patience, emotional support, encouragement, and ability to silently listen to my rants, trials, and tribulations motivated me immensely. She taught me that having a good social life is equally important to having a good academic life.

The driving factor for this research project is my advisor, Dr. Alexander Flueck, who's vision of an integrated power system dynamic simulator excited my interest in this project. I am very grateful to him for giving me the opportunity to work on this research work. Our regular discussions gave me great insight on how to approach and explore the problem in different directions. I have greatly enjoyed working with him both academically and personally.

I have greatly benefited from working with the PETSc team at Argonne National Laboratory. I want to take this opportunity to thank all the team members for developing and maintaining such a powerful numerical simulation library. In particular, I want to thank Dr. Hong Zhang for her encouragement and guidance in different facets of this project. In addition, I want to thank Dr. Barry Smith for his inputs on scientific software development and preconditioners.

I want to thank Dr. Shahidehpour, Dr. Li, Dr. Rempfer, and Dr. Zhang for being members of the thesis examining committee and providing valuable inputs during the proposal. The courses I took with Dr. Shahidehpour and Dr. Li were a great learning experience and broadened my knowledge on Power Systems and Electricity Markets.

I am grateful to Mr. Xu Zhang, Mr. Chandrahas Aserkar, and Mr. Vikram Ramanathan for their contributions in this research work.

The discussions with Dr. Gurkan Soykan, and Dr. Anurag Srivastava were of great help to me and i want to thank them for their valuable inputs.

I've had the company of some great friends in Chicago, Mumbai, and St. Louis whose friendship i've cherished the most. I want to thank each and everyone of them from the bottom of my heart.

Finally, I want to thank God Almighty for blessing me with patience and perseverance without which everything is just dust in the wind.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

Symbol	Definition
Δt_{TS}	TS time step
Δt_{EMT}	EMT time step
δ	Rotor angle
ω	Rotor speed
ω_s	Synchronous speed
$E_q^{'}$	q axis transient voltage
$E_d^{'}$	d axis transient voltage
ψ_{1d}	d axis damper winding flux
ψ_{2q}	q axis damper winding flux
n	per unit speed difference
E_{fd}	Exciter field voltage
R_F	Rate feedback
V_R	AVR voltage
s	Induction motor slip
$e_d^{'}$	Induction motor d axis transient voltage
$e_q^{'}$	Induction motor q axis transient voltage
V	Voltage magnitude
$\bar{V}$	Complex bus voltage
V_D	Real part of complex bus voltage
V_Q	Imaginary part of complex bus voltage

$V_{D,abc}$	Real part of complex bus voltage for all three phases
$V_{Q,abc}$	Imaginary part of complex bus voltage for all three phases
i_{ser}	Transmission line instantaneous series current
v	instantaneous line-ground bus voltage

ABSTRACT

The simulation of electrical power system dynamic behavior is done using transient stability simulators (TS) and electromagnetic transient simulators (EMT). A Transient Stability simulator, running at large time steps, is used for studying relatively slower dynamics e.g. electromechanical interactions among generators and can be used for simulating large-scale power systems. In contrast, an electromagnetic transient simulator models the same components in finer detail and uses a smaller time step for studying fast dynamics e.g. electromagnetic interactions among power electronics devices. Simulating large-scale power systems with an electromagnetic transient simulator is computationally inefficient due to the small time step size involved. A hybrid simulator attempts to interface the TS and EMT simulators which are running at different time steps. By modeling the bulk of the large-scale power system in a transient stability simulator and a small portion of the system in an electromagnetic transient simulator, the fast dynamics of the smaller area could be studied in detail, while providing a global picture of the slower dynamics for the rest of power system.

In the existing hybrid simulation interaction protocols, the two simulators run independently, exchanging solutions at regular intervals. However, the exchanged data is accepted without any evaluation, so errors may be introduced. While such an explicit approach may be a good strategy for systems in steady state or having slow variations, it is not an optimal or robust strategy if the voltages and currents are varying rapidly, like in the case of a voltage collapse scenario.

This research work proposes an implicitly coupled solution approach for the combined transient stability and electromagnetic transient simulation. To combine the two sets of equations with their different time steps, and ensure that the TS and EMT solutions are consistent, the equations for TS and coupled-in-time EMT equations are solved simultaneously. While computing a single time step of the TS equations, a simultaneous calculation of several time steps of the EMT equations is proposed.

Along with the implicitly coupled solution approach, this research work also proposes to use a three phase representation of the TS network instead of using a positive-sequence balanced representation as done in the existing transient stability simulators.

Furthermore a parallel implementation of the three phase transient stability simulator and the implicitly coupled electromechanical and electromagnetic transients simulator, using the high performance computing library PETSc, is presented. Results of experimentation with different reordering strategies, linear solution schemes, and preconditioners are discussed for both sequential and parallel implementation.

CHAPTER 1
INTRODUCTION

Electrical power systems are continually subjected to large disturbances, also referred to as event disturbances or contingencies, of various types such as faults, scheduled or unscheduled equipment outages, load outages, electrical or mechanical equipment failure, lightning strikes, etc. After a large disturbance, a power system may or may not return to a normal operating state, depending on the current operating state, magnitude of the disturbance, protective system operation, and preventive or corrective actions taken. Hence, credible contingencies which can cause a power system to go unstable are studied carefully.

Power system analysis is broadly classified as static and dynamic. Static analysis deals with the response to slow load/generation variations which can be studied via steady state analysis. On the other hand, large disturbance studies fall under the umbrella of dynamic analysis. Due to the multi-physics nature of the generation, transmission and load sub-systems power system dynamic phenomena range over several time scales. Electromechanical generators, which produce electricity have relatively slow mechanical dynamics, resulting in large time constants, while transmission lines and other electrical equipment have a much faster response. The different dynamic phenomena can range from microseconds to hundreds of minutes.

The analysis tools that have been developed for studying the different dynamics are specifically tailored to a particular range of time scale and are divided into two groups: Transient Stability Simulators (TS) and Electromagnetic Transients Simulators (EMT). Transient stability simulators are used for analyzing comparatively slow dynamics ranging from milliseconds to minutes, while electromagnetic transients simulators are used for faster time scales. Along with the time scale division, the modeling approach used in TS adds another fundamental difference between these two

Table 1.1. Various dynamic phenomena

Phenomenon	Timescale
Lightning propagation	Microseconds to milliseconds
Switching surges	Microseconds to tens of seconds
Electrical transients	Milliseconds to seconds
Electromechanical transients	Hundreths to tens of seconds
Mechanical transients	Tenths of seconds to hundreds of seconds
Boiler and long term dynamics	seconds to thousands of seconds

simulators. TS assumes a constant fundamental frequency of 50 or 60 Hz and represents voltages and currents as phasors. Under such an assumption, only fundamental frequency dynamics can be studied by TS. On the other hand, EMT does not use a fixed frequency assumption, so harmonics over a larger frequency spectrum (limited by the time step only) can be studied. These two simulators are briefly explained in the following sections.

1.1 Transient Stability Simulators (TS)

A transient stability simulator is an important tool for planning and design, operation and control, and post-disturbance analysis in power system [77]. It is mainly used for studying slow moving dynamics such as electromechanical generator rotor speeds. Transient stability simulators were developed to study the effects of disturbances on generator dynamics which could cause the generators to lose synchronism. The term *transient stability* in power system dynamics refers to the stability of generators following a transient. These simulators not only assess the generator stability but can also provide information about the phasor voltages and current at different buses. Transient stability simulators, in their early days, were used only to study generator dynamics and establish critical clearing times for circuit breakers. However, over the last two decades, several voltage stability incidents [47] - [73] have been reported and the role of TS to study voltage stability has grown.

The modeling of the power system equipment for TS is based on the fundamental assumption that the system frequency remains nearly constant at 50 or 60 Hz [44], depending on the local country's frequency standard. Hence, sinusoidal voltages and currents can be expressed as fundamental frequency phasor quantities. This assumption greatly enhances the capability of TS, since a large time-step can be used. Another assumption is that of balanced three phase network and operating conditions which reduces the analysis to per phase and makes the large-scale TS computational problem tractable.

The system of equations used in TS is differential-algebraic in nature, where the differential equations model dynamics of the rotating machines and the algebraic equations represent the transmission system, loads, and the connecting network. The electrical power system is expressed as a nonlinear differential-algebraic model:

$$\begin{aligned} \frac{dx}{dt} &= f(x, y, u) \\ 0 &= g(x, y) \end{aligned} \tag{1.1}$$

Using a numerical integration scheme, like the trapezoidal scheme, the differential equations are converted to algebraic equations and the two sets of nonlinear algebraic equations are solved by an iterative method such as Newton-Raphson. The time step for discretization is in the range of milliseconds. The choice of time-step, the assumptions of balanced network and constant frequency allow time-efficient simulation of large-scale power systems.

Examples of commercial packages are PSS/E, EUROSTAG, DigSilent and PowerTech TSAT.

1.2 Electromagnetic Transients Simulators (EMT)

All electrical circuits exhibit electromagnetic transients during switching. The power system with long transmission lines and various electromagnetic components exhibits very complex behavior during switching or lightning strikes. Applications such as insulation coordination, design of protection schemes, and power electronic converter design require the computation of electromagnetic transients. Such type of study is done using an electromagnetic transient simulator.

Unlike transient stability simulators, there are no assumptions on the power system to be at nearly constant 60 Hz frequency or be balanced. A full three phase representation of the power system is used in EMT. The actual current and voltage waveforms, not phasors, are used because these waveforms are of primary interest. The need for the actual voltage and current waveforms includes cases such as simulation of frequency-dependent or nonlinear components and systems, design of protection schemes, and fault analysis in series-compensated lines or high-voltage direct current lines [35].

The equations describing the power system for electromagnetic transient simulators are mostly differential, which model the generator dynamics, transmission network, connecting components, and loads. In compact form, the equations can be written as

$$\frac{dx}{dt} = f(x) \tag{1.2}$$

There also can be algebraic equations, for example modeling resistive branches, resistive faults, and circuit breakers.

The discretization time step used in EMT is on the order of microseconds (typically 50 microseconds) to capture the much faster electromagnetic transients[35]. The small time step and detailed three phase modeling requires a lot more computational effort. Practically, it is inefficient to perform electromagnetic transient analysis of large networks where all the elements are represented using detailed models[77].

Examples of commercial packages are EMTP, PSCAD/EMTDC, ATP and SimPowerSystems.

1.3 Hybrid Simulators

The ability of HVDC links to deliver large amounts of power over longer distances motivated the first efforts to do a combined AC-DC system analysis. It was realized that HVDC links could not be modeled accurately in TS [28] under faulted conditions due to rapid converter topology changes at shorter time steps and could be only studied by EMT. The need for doing an AC-DC system analysis motivated the first effort to combine TS and EMT. The introduction of power electronic flexible alternating current transmission system devices further motivated the need for interfacing the electromagnetic transient simulator with the transient stability simulator [71]. Over the years, many researchers have further explored the combined TS-EMT simulation both in terms of modeling and algorithm. *Hybrid simulator* has become a common term to refer to a combined TS-EMT simulator.

The main idea of a hybrid simulator is to split the power system into a TS region with phasor models and EMT region with detailed models. The two regions are then connected via an equivalent network of the other region and a protocol is established to transfer signals from TS to EMT and vice versa. Thus, a hybrid simulator attempts to combine the advantages of both by capturing the slow dynamics in TS and the faster dynamics in EMT along with not sacrificing computational efficiency.

1.4 Motivation

While most of the research work in developing hybrid simulators has been driven by the need to model power electronic equipment, our interest in hybrid simulators is from the dynamic security assessment point of view and the ability of the hybrid simulator to present both phasors and actual waveforms. This research work emanated from the need to capture voltage collapse trajectories and thereby differentiate between local and widespread blackouts. The prior research work done on this topic [3], [2] showed that an EMT simulation can capture the voltage collapse trajectories. Moreover, the ability to model protective relays realistically, while not sacrificing the computational speed further motivates us to explore hybrid simulators.

1.4.1 Voltage collapse.

The voltage collapse phenomenon is an important problem for electric utilities to prevent. Voltage instability incidents have been reported in power systems around the world [47], [73] and hence it becomes increasingly important to study the mechanism of voltage collapse.

Capturing the voltage collapse trajectories is important for differentiating local and widespread voltage collapses in large scale power systems. The static methods cannot capture local voltage collapse because the PV curves turn around at all the buses at the collapse point. The distance to steady state loading limit is also known as the distance to collapse and it determines the power transfer capability limit or the maximum power transfer for a given transfer direction. In contingency ranking, with respect to voltage collapse, contingencies are ranked based on their distance to collapse. The contingencies showing a small distance to collapse are ranked at the top of the contingency list and the transfer capability limit of the system corresponds to the shortest distance to collapse. An illustration of the above discussion is shown in Figure 1.1

In Figure 1.1, λ^*_{normal} is the distance to collapse with all lines in service, $\lambda^*_{ctgc_1}$ with one of the lines out and $\lambda^*_{ctgc_2}$ corresponds to the contingency having the shortest distance to collapse. The contingency with a distance to collapse $\lambda^*_{ctgc_2}$ is ranked at the top of the contingency list and the transfer capability limit for the system is

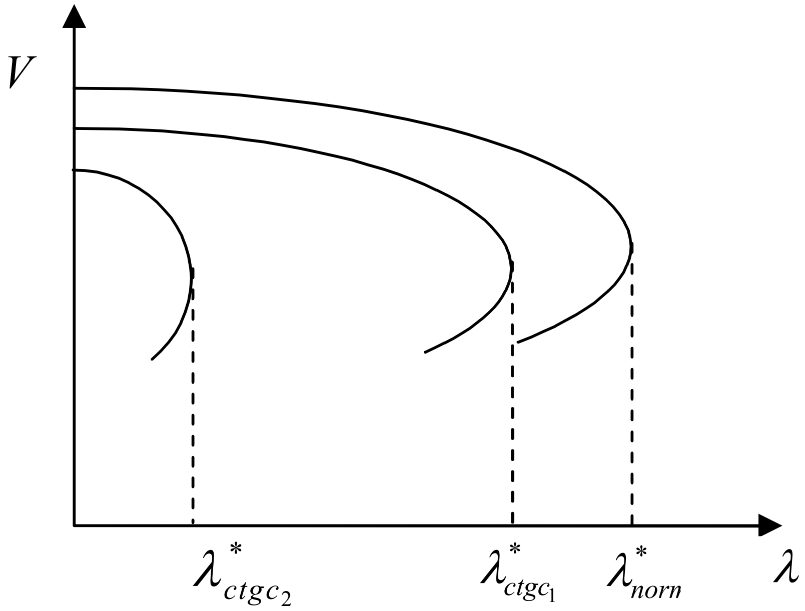


Figure 1.1. Continuation power flow curves for a system whose transfer capability limit is severely restricted by a contingency having a small distance to collapse

set to the post-ctgc steady state loading limit for this contingency. As seen, the transfer capability limit for the system is significantly reduced because of the small distance to collapse for contingency 2. However, if this particular contingency causes just a local voltage collapse then it is not a serious threat to the overall system. If contingency 2 were to occur, the load bus experiencing the localized voltage collapse would be isolated from the system by protective devices, thereby bringing the system back to a stable operating condition with increased distance to collapse. Hence, identification of such local voltage collapses is necessary for properly predicting the impact of contingencies. Moreover, a better understanding of the contingency impacts will enable the industry to better predict the transfer capability limits of large scale power systems with respect to voltage collapse.

1.4.2 Voltage collapse cascade. The voltage collapse cascade phenomenon is still a relatively unexplored domain in power system analysis. Currently, the industry predicts the potential of cascading outages based on heuristics or based on experience. However, there is no tool that can follow the sequence of events leading to a cascade or the cascading process itself including the protective device actions, for heavily loaded systems.

1.4.3 Protective system modeling in large-scale power system. The role of the protective system becomes even more important as power systems continue to operate closer to their stability limits for greater economic benefit. Protection equipment, mainly relays and circuit breakers, can isolate a faulty part of the system and thereby protect the expensive power system generation and delivery equipment from excessive currents and voltages. Setting the relay parameters for proper detection and clearing of faults is a complex operation and typically relay coordination is based on static analysis (as in traditional fault calculation)[53]. Relay settings for a longer time frame could be also determined by a TS simulation such as that needed for zone 2 or zone 3 distance relay protection[53].

However, a transient stability program cannot model and test protective relays realistically since (a) it uses fundamental frequency phasor waveforms and hence it does not have information of high frequency and/or dc signals which are presented in faulted voltages/currents, and (b) since TS uses a per phase representation of the power system network, unbalanced voltages/currents cannot be simulated. Hence conditions such as single phase operations cannot be simulated realistically.

Relay simulation is done using EMT programs using the actual current and voltage waveforms instead of phasors. Simulated or recorded faulted waveforms are used for testing the relay, and EMT programs such as EMTP are typically used. As EMT is inefficient for large-scale simulation, relay operations for only small systems can be simulated while ignoring the dynamic behavior of the rest of the system.

1.4.4 Need for a multi-timescale dynamic simulator for next-generation power grid. The electricity industry is growing through a revolution of new technologies and ideas to make the existing grid more secure, reliable and interconnected. The penetration of wind, solar, and other renewable resources of electricity production is increasing. The advent of deregulation is driving the power industry towards economic operation and thus operating the transmission system to its fullest potential. Smart Grid is bringing in a new meaning to how communication and control is done. The incorporation of power electronics equipment in power systems is increasing and brings with it non-fundamental frequency harmonics. To manage the load growth, and to enhance reliability and security, the interconnection between utility controlled transmission systems is growing. As more equipment gets added to the system and the interconnection gets denser, complex dynamic phenomenon ranging over several timescales will need to be analyzed.

1.4.5 Computational challenges for large-scale dynamic simulation. The solution of a dynamic model of a large-scale power system is computationally onerous because of the presence of a large set of DAEs that are typically stiff. Hence, dynamic analysis for large-scale systems is done off-line. Researchers at Pacific Northwest National Laboratory have reported that a simulation of 30 seconds of dynamic behavior of the Western Interconnection requires about 10 minutes of computation time today on an optimized single processor [33]. Because of this high computational cost, the dynamic analysis is only done over a small number of relatively small interconnected power system models, and computation is mainly performed off-line. However on-line dynamic analysis is needed to allow the system operators to view the system trajectories and take corrective actions before severe events cascade into a widespread blackout.

1.5 Chapter outline

Chapter 2 details the modeling and numerical solution used in transient stability simulators. Modeling of the generator, network, and loads for TS is described along with the numerical solution technique. A small example for getting some insight to the TS problem formulation is given.

The details of the EMT simulator used in this research work are described in chapter 3. The modeling of the network and loads for EMT is presented. The existing numerical solution schemes are discussed along with a small 2 bus example that presents the EMT formulation. A comparison of the simulation results for the proposed EMT simulator versus the commercial package SimPowerSystems [56] is presented.

Chapter 4 discusses the basics of hybrid simulators. Various terms used for hybrid simulators such as network equivalents, interface buses and data exchange are introduced. It presents the state of the art hybrid simulators and discusses the existing interaction protocols. Two existing interaction protocols, serial and parallel, are discussed. Finally hybrid simulation strategies for different needs are discussed.

Chapter 5 presents the motivation, details the formulation, and analyzes the results for the newly developed three phase transient stability simulator. The accuracy of the developed three phase transient stability simulator is benchmarked against the commercial package PSS/E. Results of experimentation with both direct and iterative methods as well as preconditioning schemes to speed up the three-phase TS computation on a single processor, are presented on different sized systems.

Chapter 6 details the proposed implicitly coupled TSEMT simulator and highlights the motivation and differences with the existing hybrid simulators. A novel implicitly coupled approach for combining the TS and EMT solutions at the solution phase is discussed. A new hybrid simulation termination strategy based on the phasor boundary bus voltages of EMT and TS regions is presented.

The various implementation details of the implicitly coupled TSEMT simulator such as choice of numerical integration scheme, network equivalents and disturbance simulation are detailed in Chapter 7. Results for the 9-bus and 118-bus system are presented and compared with TS and EMT. Furthermore, experimentation results with different preconditioners and reordering strategies to speed up the sequential code are presented.

Chapter 8 discusses the details of the parallel implementation of the three-phase TS and implicitly coupled TSEMT simulator. Speed up results of parallel runs on three large-sized systems with different preconditioners are presented. A novel partitioning strategy for the implicitly coupled TSEMT simulator is discussed.

The conclusions from this research work, application areas, contributions, and the future work are discussed in Chapter 9.

CHAPTER 2

TRANSIENT STABILITY SIMULATORS (TS)

Static Analysis gives a measure of the steady state operating conditions of the power system. However for events such as line outages, these methods only determine the equilibrium point of the pre-contingency and post-contingency states but do not give any information about the transient, i.e., the connecting transient state, if any, between the two steady state operating points assuming they exist, is completely ignored. A voltage collapse can occur during the transient following a contingency, so the transient response needs to be examined. One way of analyzing the transient is by observing the system trajectories in time. Transient Stability simulators with a DAE model is the typical choice for such a time domain analysis. These simulations are also called electromechanical transients simulations as they are typically used for assessing the transient stability of generators. Electromechanical transient simulators use algebraic power flow equations for the network quantities and differential equations for the generator dynamics. This differential algebraic equation model, abbreviated as DAE, and its solution methodology are discussed in this chapter. For a comprehensive discussion on transient stability simulators the reader is referred to [44].

2.1 Assumptions

To reduce the modeling complexity and thereby make the computational problem tractable certain assumptions are made for TS

1. The frequency remains nearly constant at 60 Hz [35].

2. The constant frequency assumption allows using transmission line impedances, admittances instead of using the elementary components R,L,C.

3. Voltages and current are represented using phasors, which model the fundamental frequency envelopes of the actual sinusoidal voltage and current waveforms.

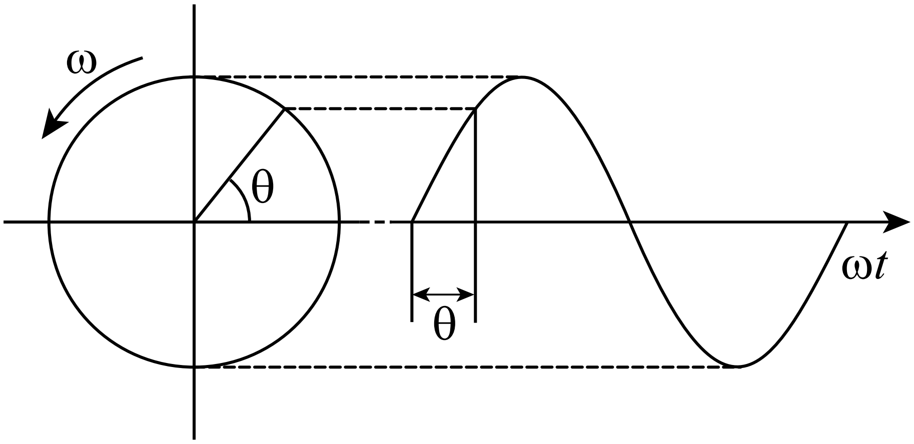


Figure 2.1. Phasor

A phasor is a representation of a sine wave with an amplitude E, phase θ and a fixed angular frequency ω. If the sinusoidal function is given by

$$e(t) = E\sin(\omega t + \theta)$$

then it can be represented in phasor form as

$$\bar{E} = Ee^{j\theta} = E_D + jE_Q$$

4. The change in network voltages and currents is instantaneous and hence a lumped transmission line model can be used.

5. The three phase network and operating conidtions are balanced at all times which enables the reduction of the three phase transmission network to single phase positive sequence network [35].

The assumption of nearly constant frequency of 60 Hz in TS allows sinusoidal voltages and currents to be represented as phasors. With phasors the network voltage and current variables are expressed either in polar form or rectangular form. This choice of variables allow TS to run at large time steps (typically 10 ms) which would be impossible if actual sinusoidal voltages and currents would be used, since 60 Hz waveforms have a period of 16.667 milliseconds. The other assumption of a balanced three-phase network reduces the size of the network system equations by a factor of three. Phasors and balanced network make a TS simulator an attractive tool for large-scale power system dynamic simulation.

2.2 Equipment modeling

An electric power system is a multi-physics system with the physics ranging from electromechanical generators to electrical transmission lines and different, diverse loads. The modeling of the power system equipment is critical to faithfully reproduce the system dynamic behavior. For studying certain dynamics, simple models might be sufficient, while for others more complex models may be needed. Various types of generator, exciter and load models have been proposed for TS while the transmission network is usually modeled by a lumped π model. This section serves to detail the different equipment modeling used in TS.

2.2.1 Generator subsystem modeling. The generator subsystem includes the electromechanical generators and the control equipment for the generators, such as exciters and turbine governors. The dynamics of the generators and associated control equipment are modeled using differential equations since they have a large time constant as compared to the electrical network. Two types of generator models with different complexities are detailed in this section.

2.2.1.1 GENROU model. This model is a three-phase round rotor generator model represented by 6 differential equations. It ignores the stator winding fluxes ψ_d and ψ_q. The model is represented in a dq machine axis reference frame and models the machine electromechanical part, rotor, and the damper winding fluxes. It assumes that there is one damper winding each present on d and q axis.

$$
\begin{aligned}
T_{d0}^{'}\frac{dE_q^{'}}{dt} &= -E_q^{'} - (X_d - X_d^{'})\bigg[I_d - \frac{X_d^{'} - X_d^{''}}{(X_d^{'} - X_l)^2} \\
&\qquad (\psi_{1d} + (X_d^{'} - X_l)I_d - E_q^{'})\bigg] + E_{fd} && (2.1) \\
T_{d0}^{''}\frac{d\psi_{1d}}{dt} &= -\psi_{1d} + E_q^{'} - (X_d^{'} - X_l)I_d && (2.2) \\
T_{q0}^{'}\frac{dE_d^{'}}{dt} &= -E_d^{'} + (X_q - X_q^{'})\bigg[I_q - \frac{X_q^{'} - X_q^{''}}{(X_q^{'} - X_l)^2} \\
&\qquad (\psi_{2q} + (X_q^{'} - X_l)I_q + E_d^{'})\bigg] && (2.3) \\
T_{q0}^{''}\frac{d\psi_{2q}}{dt} &= -\psi_{2q} - E_d^{'} - (X_q^{'} - X_l)I_d && (2.4) \\
\frac{d\delta}{dt} &= \omega_s n && (2.5) \\
2H\frac{dn}{dt} &= \frac{P_m - Dn}{1+n} - \frac{X_d^{''} - X_l}{X_d^{'} - X_l}E_q^{'}I_q - \frac{X_d^{''} - X_d^{''}}{X_d^{'} - X_l}\psi_{1d}I_q \\
&\qquad - \frac{X_q^{''} - X_l}{X_q^{'} - X_l}E_d^{'}I_d + \frac{X_q^{''} - X_q^{''}}{X_q^{'} - X_l}\psi_{2q}I_d && (2.6)
\end{aligned}
$$

Equations 2.1 - 2.4 model the electrical part of the generator while 2.5 and 2.6 model the mechanical part.

2.2.1.2 Stator equations for GENROU model. The stator equations describe the interaction of the electrical machine with the electrical network. Since the stator fluxes ψ_d and ψ_q are ignored, the stator equations become nonlinear algebraic equations.

$$
0 = -X_q^{''}I_q - \frac{X_q^{''} - X_l}{X_q^{'} - X_l}E_d^{'} + \frac{X_q^{'} - X_q^{''}}{X_q^{'} - X_l}\psi_{2q} + V_d \qquad (2.7)
$$

$$
0 = X_d^{''}I_d - \frac{X_d^{''} - X_l}{X_d^{'} - X_l}E_q^{'} + \frac{X_d^{'} - X_d^{''}}{X_d^{'} - X_l}\psi_{1d} + V_q \qquad (2.8)
$$

2.2.1.3 Generator model from [44]. This generator model is a reduced version of the GENROU model and is described by 4 differential equations. In this model the damper winding fluxes ψ_{1d} and ψ_{2q} are ignored. This generator model will be referred to as GRDC, meaning Generator Reduced, for the rest of this thesis.

$$T_{d0}^{'}\frac{dE_q^{'}}{dt} = -E_q^{'} - (X_d - X_d^{'})I_d + E_{fd} \tag{2.9}$$

$$T_{q0}^{'}\frac{dE_d^{'}}{dt} = -E_d^{'} + (X_q - X_q^{'})I_q \tag{2.10}$$

$$\frac{d\delta}{dt} = \omega - \omega_s \tag{2.11}$$

$$\frac{2H}{\omega_s}\frac{d\omega}{dt} = T_M - E_d^{'}I_d - E_q^{'}I_q - (X_q^{'} - X_d^{'})I_dI_q - D(\omega - \omega_s) \tag{2.12}$$

2.2.1.4 Stator equations for generator model from [44]. The stator algebraic equations for the GRDC model are

$$0 = E_d^{'} - V_d - R_sI_d + X_q^{'}I_q \tag{2.13}$$

$$0 = E_q^{'} - V_q - R_sI_d - X_d^{'}I_d \tag{2.14}$$

2.2.1.5 IEEE type 1 exciter model. IEEE type 1 exciter model (or IEEET1) is a third order exciter model which describes the dynamics of the exciter, rate feedback loop and the automatic voltage regulator. The IEEE type 1 exciter model in PSS/E has one additional differential equation for the voltage transducer. This is ignored in [44] and also not implemented in TSEMT.

$$T_E\frac{dE_{fd}}{dt} = -(K_E + S_E(E_{fd}))E_{fd} + V_R \tag{2.15}$$

$$T_F\frac{dR_F}{dt} = -R_F + \frac{K_F}{T_F}E_{fd} \tag{2.16}$$

$$T_A\frac{dV_R}{dt} = -V_R + K_AR_F - \frac{K_AK_F}{T_F}E_{fd} + K_A(V_{ref} - V) \tag{2.17}$$

The above model is combined with limits on the automatic voltage regulator output: $V_{Rmin} \leq V_R \leq V_{Rmax}$. The saturation function $S_E(E_{fd})$ differs in PSS/E and [44]. [44] uses an exponential form of saturation function $S_E(E_{fd}) = Ae^{BE_{fd}}$ while PSS/E uses a quadratic saturation function $S_E(E_{fd}) = B(E_{fd} - A)^2/E_{fd}$.

2.2.2 Machine to network transformation. The electrical machine equations are typically represented on a rotating(rotor) dq axis reference frame. This reference frame allows the elimination of time-varying inductances by referring the stator and rotor quantities on a rotating reference frame. In the case of a synchronous machine, the stator quantities are referred to the rotor. I_d and I_q represent the two DC currents flowing in the two equivalent rotor windings (d winding directly on the same axis as the field winding, and q winding on the quadratic axis), producing the same flux as the stator I_a, I_b, and I_c currents. The machine-network transformation for TS is given by[44]

$$\begin{bmatrix} V_d \\ \\ V_q \end{bmatrix} = \begin{bmatrix} \sin\,\delta & -\cos\,\delta \\ \\ \cos\,\delta & \sin\,\delta \end{bmatrix} \begin{bmatrix} V_{genD} \\ \\ V_{genQ} \end{bmatrix} \tag{2.18}$$

where the complex voltage at the generator bus in rectangular coordinates is $\bar{V}_{gen} = V_{genD} + jV_{genQ}$. Likewise, the current transformation is as follows

$$\begin{bmatrix} I_{genD} \\ \\ I_{genQ} \end{bmatrix} = \begin{bmatrix} \sin\,\delta & -\cos\,\delta \\ \\ \cos\,\delta & \sin\,\delta \end{bmatrix} \begin{bmatrix} I_d \\ \\ I_d \end{bmatrix} \tag{2.19}$$

2.2.3 Network subsystem modeling. The modeling of the transmission network in transient stability simulators is the same as that done for steady state analysis. This is due to the *quasi steady-state* assumption used in transient stability analysis which assumes that the changes in network voltages and currents are very fast compared to the dynamics of the rotating machines. Hence, a steady state equivalent model for the transmission network can be used. The equations for the network can be expressed either in current balance or power balance form. A current balance form representation of network equations is preferred over the power balance form for the numerical solution process [44]. The network equations are represented in complex current balance form as

$$\bar{Y}_{bus}\bar{V} = \bar{I}_{inj} \tag{2.20}$$

$\bar{I}_{inj}$ is the vector of the sum of complex generator current $\bar{I}_{gen}$ and load current $\bar{I}_{load}$ injected into the network nodes.

Splitting the complex voltage vector $\bar{V}$ into real and imaginary parts V_D and V_Q, equation 2.20 can be written as

$$\begin{bmatrix} G & -B \\ B & G \end{bmatrix} \begin{bmatrix} V_D \\ V_Q \end{bmatrix} = \begin{bmatrix} I_{Dinj} \\ I_{Qinj} \end{bmatrix} \tag{2.21}$$

In equation 2.21, G and B are the real and imaginary parts of the complex $\bar{Y}_{bus}$ matrix. If the voltage variables are arranged as $V = [V_{D1}, V_{Q1}, V_{D2}, V_{Q2}, \ldots, V_{Dn}, V_{Qn}]^t$, then using this ordering the "Y" matrix in 2.21 becomes a matrix with a 2X2 block for each branch connection and for each bus on the diagonal.

2.2.4 Load subsystem modeling. The modeling of load is a complex task for power system planners and operators because of the load diversity and scale. It is impossible and computationally infeasible to model each and every load element beginning from household appliances to industry loads. As transient stability simulators are used for transmission network studies primarily individual loads are lumped together at substation buses and their net effect is represented by different types of load models.

There are two classes of load models used for transient stability studies *static* and *dynamic* loads. Static load models are described in terms of linear or nonlinear functions of bus voltage while dynamic loads use differential equations to model the load dynamics. Depending on the observed load characteristic the load model at any particular bus is developed and either a static, dynamic or a combination of static and dynamic load models can be used. The load models used in this work are described next.

2.2.4.1 Static load models. These types of load models describe the relationship between the bus currents and the load power as a function of bus voltage. Static loads can be modeled in TS with the network in current balance form as

$$\bar{I}_{load} = \frac{(P_0 - jQ_0)}{V^2}\bar{V} \tag{2.22}$$

Here P_0, Q_0 are the initial load real and reactive powers, V is the load bus voltage magnitude while $\bar{V}$ is the complex bus voltage. In a transient stability simulation with constant power load models, V and $\bar{V}$ correspond to the current time instant. For a constant impedance load, V is held constant for all time steps.

2.2.4.2 Induction motor [52]. Induction motors are widely used in industries as well as household appliances. The modeling of the induction motors is done by either steady state induction motor load model or dynamic load model described by differential equations. The dynamic load model for a single cage induction motor is given by

$$\frac{de_d^{'}}{dt} = \omega_s s e_q^{'} - \left(e_d^{'} + (X_0 - X^{'})I_q\right)/T_0^{'} \tag{2.23}$$

$$\frac{de_q^{'}}{dt} = \omega_s s e_d^{'} - \left(e_q^{'} + (X_0 - X^{'})I_d\right)/T_0^{'} \tag{2.24}$$

$$2H\frac{ds}{dt} = T_M(s) - T_E \tag{2.25}$$

where the electrical torque is

$$T_E \approx e_d^{'}I_d + e_q^{'}I_q$$

and X_0,$X^{'}$ and T_0 can be derived from the motor parameters

$$X_0 = X_s + X_m$$
$$X_i = X_s + \frac{X_r X_m}{X_r + X_m}$$
$$T_0^{'} = \frac{X_r + X_m}{\omega_s R_r}$$

Equations 2.23 - 2.25 describe the differential equations for a voltage behind the stator resistance R_s and the motor slip. The mechanical torque T_M is a function of the motor slip and different models have a different expression for T_M. The induction motor model CIM5BL [54] uses mechanical torque of the form

$$T_M = T_{M0}(1+s)^D$$

where T_{M0} is the initial mechanical torque and D is the damping coefficient. This induction motor model is implemented in TS3ph and TSEMT.

2.2.5 Fault modeling. Power systems undergo disturbances of various sorts such as balanced and unbalanced faults, equipment outage of generators, transmission lines and other equipment, unnecessary breaker trippings, etc. The goal of the transient stability simulators is to determine whether the power system recovers following a disturbance and hence disturbance modeling is an important issue. A common disturbance simulation involves placing a fault at a given node at some prespecified time and removing the fault by opening a circuit element at another prespecified time. This type of disturbance scenario can help determine the cricitial clearing time of the circuit breakers and thereby protect the electrical machines from going out of synchronism.

A fault is modeled typically in TS by adding a large shunt conductance at the given faulted node. This large shunt conductance represents a low resistance path to ground for the faulted node and thus large currents, as seen during faulted conditions, can be modeled. PSS/E uses this type of modeling for faults.

2.3 Equations and variables

Using a current balance form for the network equations and a rectangular form for the network voltages and currents, the equations for TS are

$$\frac{dx_{gen}}{dt} = f(x_{gen}, I_{dq}, V_{DQ}) \tag{2.26}$$

$$0 = h(x_{gen}, I_{dq}, V_{DQ}) \tag{2.27}$$

$$\begin{bmatrix} G & -B \\ \\ B & G \end{bmatrix} \begin{bmatrix} V_D \\ \\ V_Q \end{bmatrix} = \begin{bmatrix} I_{gen,D}(x_{gen}, I_{dq}) \\ \\ I_{gen,Q}(x_{gen}, I_{dq}) \end{bmatrix} - \begin{bmatrix} I_{load,D}(x_{load}, V_{DQ}) \\ \\ I_{load,Q}(x_{load}, V_{DQ}) \end{bmatrix} \tag{2.28}$$

$$\frac{dx_{load}}{dt} = f_2(x_{load}, V_{DQ}) \tag{2.29}$$

Grouping all the dynamic variables together in one set and all the algebraic variables in another set, the TS equations can be described by the differential-algebraic model

$$\begin{aligned} \frac{dx}{dt} &= f(x, y, u) \\ 0 &= g(x, y) \end{aligned} \tag{2.30}$$

here

$$\begin{aligned} x &\equiv [x_{gen}, x_{load}]^t \\ y &\equiv [I_d, I_q, V_D, V_Q]^t \end{aligned}$$

x_{gen} are the dynamic variables for the generator subsystem i.e. the generator, exciter, and turbine governor dynamic variables. x_{gen} for each generator varies depending on the generator, exciter, turbine governor and other control equipment models. For a generator modeled using a GENROU model, and an IEEET1 exciter model, then x_{gen} is as follows:

$$x_{gen} \equiv \left[E_q^{'}, E_d^{'}, \psi_{1d}, \psi_{2q}, \delta, n, E_{fd}, R_F, V_R\right]^t$$

The number of variables in the x_{load} equals the number of dynamic load variables. If there are no dynamic load variables then the size of the x_{load} vector is 0. For an induction motor model, the dynamic variables are $x_{load} = \left[e_q^{'}, e_d^{'}, s\right]^t$.

2.3.1 Disturbance simulation. Typical large disturbances include faults on the network, line trippings, generator outages, load outages etc. Such disturbances are very fast compared to the generator dynamics which have large mechanical time constants. The stator equations are also electrical equations and are assumed to respond instantaneously to the disturbance. Hence, the network and the stator algebraic variables are solved at the disturbance time to reflect the post-disturbance values. This one additional solution at the disturbance time involves the solution of the equations

$$I_{dq}(t_d+) = h^f\left(x(t_d), \bar{V}(t_d+)\right) \tag{2.31}$$

$$0 \qquad = g^f\left(x(t_d), I_{dq}(t_d+), \bar{V}(t_d+)\right) \tag{2.32}$$

where the superscript$f$ indicates that the algebraic equations correspond to the faulted state and t_d represents the fault time. With the post-disturbance algebraic solution thus obtained, the numerical integration process is again resumed.

2.4 Two bus system example

This example serves as a simple example to detail the different variables and equations used in TS. Figure2.2 shows a two-bus system having one transmission line connecting a generator at bus 1 to a load at bus 2. Assume, for the sake of simplicity, that the generator model is GRDC without any exciter or turbine governor model and the bus 2 load model is constant impedance. The network voltages are represented

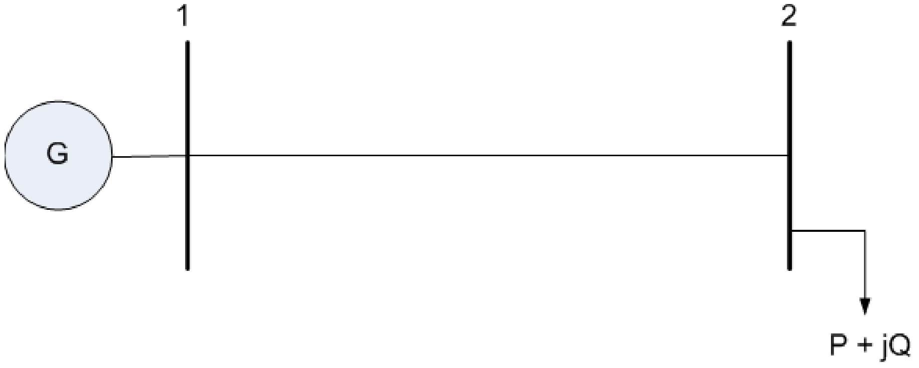


Figure 2.2. One line diagram of the two bus example system

in rectangular form V_D and V_Q. The equations for the generator subsystem are

$$T_{d0}^{'}\frac{dE_q^{'}}{dt} = -E_q^{'} - (X_d - X_d^{'})I_d + E_{fd} \tag{2.33}$$

$$T_{q0}^{'}\frac{dE_d^{'}}{dt} = -E_d^{'} + (X_q - X_q^{'})I_q \tag{2.34}$$

$$\frac{d\delta}{dt} = \omega - \omega_s \tag{2.35}$$

$$\frac{2H}{\omega_s}\frac{d\omega}{dt} = T_M - E_d^{'}I_d - E_q^{'}I_q - (X_q^{'} - X_d^{'})I_dI_q - D(\omega - \omega_s) \tag{2.36}$$

with the stator algebraic equations

$$0 = E_d^{'} - V_d - R_sI_d + X_q^{'}I_q \tag{2.37}$$

$$0 = E_q^{'} - V_q - R_sI_d - X_d^{'}I_d \tag{2.38}$$

The stator algebraic equations need V_d and V_q which is obtained from doing a transformation of bus 1 voltages V_{1D} and V_{1Q} to the synchronous rotating frame of the generator.

$$\begin{bmatrix} V_d \\ \\ V_q \end{bmatrix} = \begin{bmatrix} \sin\ \delta & -\cos\ \delta \\ \\ \cos\ \delta & \sin\ \delta \end{bmatrix} \begin{bmatrix} V_{1D} \\ \\ V_{1Q} \end{bmatrix}$$

The generator current injection at bus 1 is

$$\begin{bmatrix} I_{genD} \\ \\ I_{genQ} \end{bmatrix} = \begin{bmatrix} \sin\ \delta & -\cos\ \delta \\ \\ \cos\ \delta & \sin\ \delta \end{bmatrix} \begin{bmatrix} I_d \\ \\ I_d \end{bmatrix}$$

Assuming that the constant impedance load at bus 2 is drawing apparent power $P + jQ$ at steady state with steady state bus voltage is V_{m0}, then the current drawn by the load in rectangular form is

$$\begin{aligned} I_{loadD} &= \frac{P}{V_{m0}^2}V_{2D} + \frac{Q}{V_{m0}^2}V_{2Q} \\ I_{loadQ} &= -\frac{Q}{V_{m0}^2}V_{2D} + \frac{P}{V_{m0}^2}V_{2Q} \end{aligned}$$

The relationship between the complex voltages and currents for this 2 bus system is given by the nodal network equation

$$\begin{bmatrix} \bar{Y}_{11} & \bar{Y}_{12} \\ \bar{Y}_{21} & \bar{Y}_{22} \end{bmatrix} \begin{bmatrix} \bar{V}_1 \\ \bar{V}_2 \end{bmatrix} = \begin{bmatrix} \bar{I}_{1inj} \\ \bar{1}_{2inj} \end{bmatrix}$$

Writing the nodal network equation in rectangular form and including the generator and load current injections, the algebraic equations to be solved for the network are

$$\begin{bmatrix} G_{11} & -B_{11} & G_{12} & -B_{12} \\ B_{11} & G_{11} & B_{12} & G_{12} \\ G_{21} & -B_{21} & G_{22} & -B_{22} \\ B_{21} & G_{21} & B_{22} & G_{22} \end{bmatrix} \begin{bmatrix} V_{1D} \\ V_{1Q} \\ V_{2D} \\ V_{2Q} \end{bmatrix} = \begin{bmatrix} I_{genD} \\ I_{genQ} \\ -I_{loadD} \\ -I_{loadQ} \end{bmatrix} \tag{2.39}$$

Equations 2.33-2.39 are the equations which can be represented in differential-algebraic form as given by equation 2.30 with the variables $x \equiv [E_q^{'}, E_d^{'}, \Delta, \omega]^t$ and $y \equiv [I_d, I_q, V_{1D}, V_{1Q}, V_{2D}, V_{2Q}]^t$.

2.5 Numerical solution of TS equations

The differential equations in 2.30 are discretized using a numerical integration scheme. The most used numerical integration scheme is the implicit trapezoidal integration scheme because of its simplicity and numerical A-stability properties[44]. Using the implicit trapezoidal scheme, the equations to be solved by TS are

$$\begin{aligned} x(t+\Delta t) - x(t) - \frac{\Delta t}{2}\left(f(x(t+\Delta t), y(t+\Delta t)) + f(x(t), y(t))\right) = 0 \\ g(x(t+\Delta t, y(t+\Delta t))) = 0 \end{aligned} \tag{2.40}$$

Equation 2.40 is then solved iteratively using Newton's method. One of the prefered methods to solve the linear system in Newton's method is to use a Schur complement solution process. The linear system to be solved at each newton iteration is given by

$$\begin{bmatrix} J_{xx} & J_{xV} \\ \\ J_{Vx} & J_{VV} \end{bmatrix} \begin{bmatrix} \Delta x \\ \\ \Delta V \end{bmatrix} = - \begin{bmatrix} F_x \\ \\ F_V \end{bmatrix} \tag{2.41}$$

The solution process is done in two steps where the network voltages are solved first

$$\left(J_{VV} - J_{Vx}J_{xx}^{-1}J_{xV}\right)\Delta V = -F_V + J_{Vx}J_{xx}^{-1}F_V \tag{2.42}$$

and then Δx is computed by solving the linear equation

$$J_{xx}\Delta x = -F_x \tag{2.43}$$

It is to be noted here that the x vector includes all the variables for the generator subsystem, i.e., it includes the stator currents I_{dq} variables.

The advantage of the Schur complement method for solution is that the J_{xx} submatrix is a block-diagonal matrix and its inverse can be found easily. Moreover the matrix $J_{VV} - J_{Vx}J_{xx}^{-1}J_{xV}$ has the same sparsity pattern as J_{VV} and hence it need be symbolically factored only once.

CHAPTER 3
ELECTROMAGNETIC TRANSIENT SIMULATORS (EMT)

All electrical circuits exhibit electromagnetic transients during switching. The power system with long transmission lines and various electromagnetic components exhibits very complex behavior during switching or lightning strikes. Applications such as insulation coordination, design of protection schemes, and power electronic converter design require the computation of electromagnetic transients. Such type of study is done using an electromagnetic transient simulator. Unlike transient stability simulators, there are no assumptions for the power system to be at nearly constant 60 Hz frequency or be balanced. A full three phase representation of the power system is used. The actual current and voltage waveforms are used because these waveforms are of primary interest. The need for the analyzing actual voltage and current waveforms includes cases such as simulation of frequency-dependent or nonlinear components and systems, design of protection schemes, and fault analysis in series-compensated lines or HVDC lines[35]. The equations describing the power system for electromagnetic transient simulators are mostly differential, which model the generator dynamics, transmission network, connecting components, and loads. In compact form, the equations can be written as

$$\frac{dx_{EMT}}{dt} = f(x_{EMT}) \tag{3.1}$$

There also can be algebraic equations too, for example modeling resistive branches, resistive faults, and circuit breakers. Like transient stability simulators, a discretization technique such as the trapezoidal rule is used to convert the differential equations into algebraic equations. The new set of non-linear equations is solved using an iterative scheme like Newton-Raphson. The discretization time step is on the order of microseconds (typically 50 microseconds) to capture the fast electromagnetic transients. The small time step and detailed three phase modeling requires a lot more computational effort than a TS simulation. Practically, it is inefficient to perform electromagnetic transient analysis of large networks where all the elements are represented using detailed models[77]. Examples of commercial packages are EMTP, PSCAD/EMTDC, ATP and SimPowerSystems.

3.1 Equipment modeling

3.1.1 Generator subsystem modeling. The modeling of the electrical generators, exciters, and turbine governors is similar to the detailed modeling in the generator subsystem modeling section for transient stability simulators in subsection 2.2.1. Since EMT uses a much smaller time step for simulation, more detailed generator models can be modeled along with their complex dynamics which can be captured only at the electromagnetic time-scale. The only difference in the modeling is the machine-network transformation. Since EMT uses instantaneous voltages, the three phase instantaneous values need to be converted to dq quantities and vice versa. The machine-network transformation is given by

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \sin(\theta) & \sin(\theta - 2\pi/3) & \sin(\theta + 2\pi/3) \\ \cos(\theta) & \cos(\theta - 2\pi/3) & \cos(\theta + 2\pi/3) \end{bmatrix} \begin{bmatrix} v_{gen,a} \\ v_{gen,b} \\ v_{gen,c} \end{bmatrix} \tag{3.2}$$

and

$$\begin{bmatrix} i_{gen,a} \\ i_{gen,b} \\ i_{gen,c} \end{bmatrix} = \begin{bmatrix} \sin(\theta) & \cos(\theta) \\ \sin(\theta - 2\pi/3) & \cos(\theta - 2\pi/3) \\ \sin(\theta + 2\pi/3) & \cos(\theta + 2\pi/3) \end{bmatrix} \begin{bmatrix} I_d \\ I_q \end{bmatrix} \tag{3.3}$$

Assuming that the q axis is aligned with phase a axis in steady state, θ equals $\delta - (\pi/2)\omega_s t$

3.1.2 Network subsystem modeling. The network subsystem includes the electrical transmission lines, transformers, shunt capacitors and other associated circuitry. EMT simulations are carried out to study complex high frequency phenomena on the transmission lines such as the effect of lightning strikes, capacitor switching, surge overvoltages etc. Hence EMT uses distributed parameter lines or frequency dependent models of the transmission line which can describe the dynamics over a larger frequency range. For the scope of this thesis, distributed parameter or frequency dependent transmission line models are not used and transmission lines are modeled using lumped equivalent π models. This simplification is justified for short and medium transmission lines and is used to ease the implementation of the implicitly coupled TSEMT simulator. The following subsection explains the lumped equivalent transmission line model for EMT.

3.1.2.1 Lumped π model transmission line. A lumped model transmission line is modeled by lumping the distributed parameters. If, $R^{'}$, $L^{'}$ and $C^{'}$ are the distributed parameters of a line of length d, then the lumped parameters are obtained by multiplying the distributed parameters with the line length. The relationship between the currents and the voltages at the two ends is obtained by using KCL at the line ends and KVL for the series branch.

$$L\frac{di_{ser}}{dt} = v_k(t) - v_m(t) - Ri_{ser}(t) \tag{3.4}$$

$$\frac{C}{2}\frac{dv_k}{dt} = i_{km}(t) - i_{ser}(t) \tag{3.5}$$

$$\frac{C}{2}\frac{dv_m}{dt} = i_{ser}(t) + i_{mk}(t) \tag{3.6}$$

here R, L, C are the lumped parameters of the transmission line.

The formulation displayed in equations 3.4-3.6 can be extended to the modeling of three phase transmission lines. The equations for the three phase π model

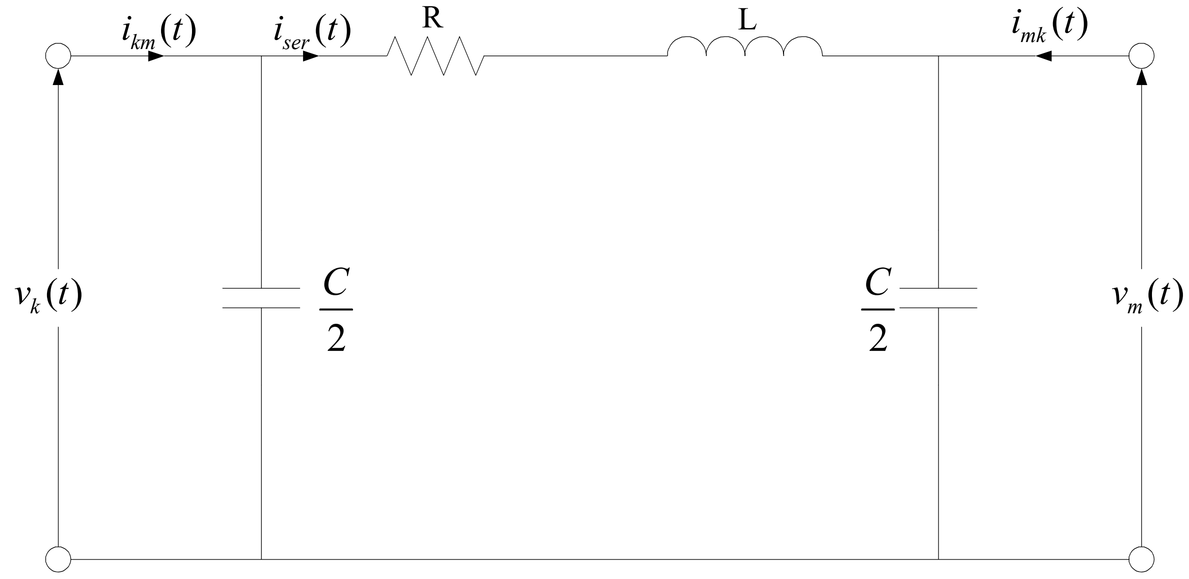


Figure 3.1. Lumped π model of a transmission line

transmission line can be written as

$$[L]\frac{di_{ser,abc}}{dt} = v_{k,abc}(t) - v_{m,abc}(t) - [R]i_{ser,abc}(t) \tag{3.7}$$

$$\frac{[C]}{2}\frac{dv_k, abc}{dt} = i_{km,abc}(t) - i_{ser,abc}(t) \tag{3.8}$$

$$\frac{[C]}{2}\frac{dv_m, abc}{dt} = i_{ser,abc}(t) + i_{mk,abc}(t) \tag{3.9}$$

Here $[R]$, $[L]$, and $[C]$ are 3X3 matrices having the following self and the mutual elements

$$[R] \equiv \begin{bmatrix} R_{aa} & R_{ab} & R_{ac} \\ R_{ba} & R_{bb} & R_{bc} \\ R_{ca} & R_{cb} & R_{cc} \end{bmatrix}, \ [L] \equiv \begin{bmatrix} L_{aa} & L_{ab} & L_{ac} \\ L_{ba} & L_{bb} & L_{bc} \\ L_{ca} & L_{cb} & L_{cc} \end{bmatrix}, \ [C] \equiv \begin{bmatrix} C_{aa} & C_{ab} & C_{ac} \\ C_{ba} & C_{bb} & C_{bc} \\ C_{ca} & C_{cb} & C_{cc} \end{bmatrix}$$

3.1.3 Load modeling.

The loads for power system steady state analysis are represented by real power P and reactive power Q. Since P and Q are applicable to only phasor domain and cannot be directly represented in instantaneous domain, real and reactive power loads are modeled using resistances and inductances. The resistance R_{load} and the inductance L_{load} of a load drawing real power P and reactive power Q is given by

$$R_{load} = \frac{V_{m0}^2}{P} \tag{3.10}$$

$$L_{load} = \frac{V_{m0}^2}{\omega Q} \tag{3.11}$$

where $\omega = 2\pi f$ and f is the fundamental frequency. The load branch can be connected either in series or in parallel configuration and a three phase load can be represented by having a series or parallel branch on each phase. The loads modeled in the developed EMT simulator and TSEMT all use parallel RL branches. Using series RL loads is one of the future work topics. Modeling loads by series or parallel branches results in different dynamics since for a series RL load the entire load current cannot change instantaneously while for a parallel RL load only a part of the load current (flowing through the inductor branch) cannot change instantaneously.

3.1.3.1 Modeling of constant power loads [3], [2]. Constant power loads for steady state studies are modeled by fixed negative power injections into the network. For constant power loads there is no dependence of voltage. Such modeling is based on the time period of interest involved in steady state studies. Physically however, any device can be thought of as sensing the stimulus first, before reacting to it. Thus, a load having constant power characteristics reacts to a change in the voltage or current after sensing it first. The quicker it responds, the closer it is to absorbing constant power during the transient state. Our modeling of loads trying to absorb constant power is based on this notion. Constant power loads are modeled by real and reactive power absorbed through a parallel shunt which is not a constant shunt value but rather a time varying shunt that depends on the previous cycle of the fundamental frequency voltage waveform.

For a constant impedance load, if the voltage magnitude decreases, then the current drawn decreases and the load power decreases too, since the load impedance is constant. For a load absorbing constant power however, if the voltage magnitude decreases, then the current increases to maintain constant power. This increase in the current as the voltage decreases can be modeled by decreasing the load impedance as the voltage decreases. Using equation 3.13, loads trying to absorb constant power are modeled by changing the resistance and the inductance of the load at each time step. This requires the knowledge of the voltage magnitude V at each time step. A fourier analysis of the voltage waveform over the previous cycle of the fundamental frequency gives the voltage magnitude at each time instant.

If n and $n-1$ represent the t_n and t_{n-1} time instants, the resistance and inductance are modified at each time step as

$$R_{load,n} = \frac{V_{n-1}^2}{P} \tag{3.12}$$

$$L_{load,n} = \frac{V_{n-1}^2}{\omega Q} \tag{3.13}$$

here, the voltage magnitude V_{n-1} is calculated by doing a fourier analysis over a running window of one cycle of fundamental frequency. The subscript $n-1$, associated with the voltage magnitude, indicates that the instantaneous voltages over one cycle of fundamental frequency ending at the $n-1$ time instant are used to calculate the load shunt values at time instant n. Thus, the load responds to the voltage magnitude from the previous time instant. Since time step for EMT being very small, it can be assumed that the load responds almost instantaneously, thus mimicking a constant power load.

3.2 Equations and variables

Simulation of large interconnected systems requires information about which equipment is incident on which nodes. Incidence matrices, which map the connection between elements serve this purpose.The definition of a few incidence matrices that describe the mapping of current injections onto the network is described as follows:

- $A_{gen} \equiv$ Incidence matrix for generator current injections onto the network nodes.
- $A_{load} \equiv$ Incidence matrix for load current injections onto the network nodes.
- $A_{ser} \equiv$ Incidence matrix for mapping the transmission line current injections.
- $A_{fault} \equiv$ Incidence matrix for mapping the fault currents.

Given the above incidence matrix definitions, the equations describing the power system for EMT are

$$\frac{dx_{gen}}{dt} = f(x_{gen}, I_{dq}) \tag{3.14}$$

$$0 = h(I_{dq}, x_{gen}, v) \tag{3.15}$$

$$[L]\frac{di_{ser}}{dt} = -A_{ser}^t v - [R]i_{ser} \tag{3.16}$$

$$\frac{[C]}{2}\frac{dv}{dt} = A_{gen}i_{gen}(I_{dq}) - A_{load}i_{load}(x_{load}, v) + A_{ser}i_{ser} - A_{fault}i_{fault} \tag{3.17}$$

$$0 = A_{fault}^t v - [R_{fault}]i_{fault} \tag{3.18}$$

$$\frac{dx_{load}}{dt} = g(x_{load}, v) \tag{3.19}$$

Here the definitions of variables x_{gen} and I_{dq} follow that from section 2.3, i_{ser} is the vector of three phase instantaneous transmission line currents, v is the three phase instantaneous node voltage vector. x_{load} are the load variables which could include the currents through the load inductors, if the loads are modeled as series or parallel branches. For induction motors, x_{load} includes the dynamic variables for the induction motor.

3.2.1 An illustrative example describing the different equations and variables.

The same two-bus system example from chapter 2, Figure 2.2, is used as an example system. Assume that the load at bus 2 is modeled by parallel RL branches.

The transmission line 1-2 is modeled by a lumped π model line. For the sake of keeping the equations simple, assume that the generator model is GRDC without any exciter or turbine governor. The equations describing this system are

$$T_{d0}^{'}\frac{dE_q^{'}}{dt} = -E_q^{'} - (X_d - X_d^{'})I_d + E_{fd} \tag{3.20}$$

$$T_{q0}^{'}\frac{dE_d^{'}}{dt} = -E_d^{'} + (X_q - X_q^{'})I_q \tag{3.21}$$

$$\frac{d\delta}{dt} = \omega - \omega_s \tag{3.22}$$

$$\frac{2H}{\omega_s}\frac{d\omega}{dt} = T_M - E_d^{'}I_d - E_q^{'}I_q - (X_q^{'} - X_d^{'})I_dI_q - D(\omega - \omega_s) \tag{3.23}$$

with the stator algebraic equations

$$0 = E_d^{'} - V_d - R_sI_d + X_q^{'}I_q \tag{3.24}$$

$$0 = E_q^{'} - V_q - R_sI_d - X_d^{'}I_d \tag{3.25}$$

For the stator algebraic equations, bus 1,voltages need to be converted from the network *abc* reference frame to machine *dq* axis reference frame as follows:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \sin(\theta) & \sin(\theta - 2\pi/3) & \sin(\theta + 2\pi/3) \\ \cos(\theta) & \cos(\theta - 2\pi/3) & \cos(\theta + 2\pi/3) \end{bmatrix}\begin{bmatrix} v_{1a} \\ v_{1b} \\ v_{1c} \end{bmatrix}$$

The generator current injection i_{gen} at bus 1 is given by

$$\begin{bmatrix} i_{gen,a} \\ i_{gen,b} \\ i_{gen,c} \end{bmatrix} = \begin{bmatrix} \sin(\theta) & \cos(\theta) \\ \sin(\theta - 2\pi/3) & \cos(\theta - 2\pi/3) \\ \sin(\theta + 2\pi/3) & \cos(\theta + 2\pi/3) \end{bmatrix}\begin{bmatrix} I_d \\ I_q \end{bmatrix}$$

For a parallel three phase *RL* load the load current injection i_{load} equals the sum of the current flowing through the resistor and the current through the inductor.

$$\begin{bmatrix} i_{load,a} \\ i_{load,b} \\ i_{load,c} \end{bmatrix} = \begin{bmatrix} v_{2a}/R_{loada} + i_{Lloada} \\ v_{2b}/R_{loadb} + i_{Lloadb} \\ v_{2c}/R_{loadc} + i_{Lloadc} \end{bmatrix}$$

The different incidence matrices for this illustrative system are

$$A_{gen} \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, A_{load} \equiv \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, A_{ser} \equiv \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

In A_{ser}, the current sign convention is assumed as current going out of the node is a negative current injection.

Since there is one transmission line and two buses, there will be 3 differential equations for the transmission line series current (one for each phase) and 6 differential equations for the bus voltages. Before writing the equations for the network, let us define the $[R]$, $[L]$, and $[C]$ matrices

$$[R] \equiv \begin{bmatrix} R_{12aa} & R_{12ab} & R_{12ac} \\ R_{12ba} & R_{12bb} & R_{12bc} \\ R_{12ca} & R_{12cb} & R_{12cc} \end{bmatrix}, \quad [L] \equiv \begin{bmatrix} L_{12aa} & L_{12ab} & L_{12ac} \\ L_{12ba} & L_{12bb} & L_{12bc} \\ L_{12ca} & L_{12cb} & L_{12cc} \end{bmatrix}$$

$$[C] \equiv \begin{bmatrix} C_{12aa} & C_{12ab} & C_{12ac} & & & \\ C_{12ba} & C_{12bb} & C_{12bc} & & & \\ C_{12ca} & C_{12cb} & C_{12cc} & & & \\ & & & C_{12aa} & C_{12ab} & C_{12ac} \\ & & & C_{12ba} & C_{12bb} & C_{12bc} \\ & & & C_{12ca} & C_{12cb} & C_{12cc} \end{bmatrix}$$

The diagonal elements in each 3 x 3 block in the above matrices are the self elements and the off-diagonal elements represent the coupling with other phases. Defining the vector $i_{ser} \equiv [i_{sera}, i_{sera}, i_{serc}]^t$ and the bus voltage vector $v \equiv [v_{1a}, v_{1b}, v_{1c}, v_{2a}, v_{2b}, v_{2c}]^t$, the differential equations for the π transmission line model can be written as

$$[L]\frac{di_{ser}}{dt} = - A^t_{ser}v - [R]i_{ser} \tag{3.26}$$

$$\frac{[C]}{2}\frac{dv}{dt} = A_{gen}i_{gen}(I_{dq}) - A_{load}i_{load}(x_{load}, v) + A_{ser}i_{ser} \tag{3.27}$$

And finally the equations for the current through the load inductor branch are

$$[L_{load}]\frac{d_{iLload}}{dt} = v_2 \tag{3.28}$$

Here i_{Lload} is the load inductor current for the three phases, $[L_{load}]$ is the diagonal matrix of load inductances, and v_2 represents bus 2 phase abc voltages.

Equations 3.20-3.28 describe the equations for the different dynamics of this example two bus system. The variables to be solved at each time step are

$$x \equiv [E^{'}_q, E^{'}_d, \delta, \omega, I_d, I_q, i_{sera}, i_{serb}, i_{serc}, v_{1a}, v_{1b}, v_{1c}, v_{2a}, v_{2b}, v_{2c}, i_{Lloada}, i_{Lloadb}, i_{Lloadc}]^t$$

3.3 Numerical solution

The numerical solution of EMT equations can be either done doing state variable analysis or by numerical integration substitution method [35]. In the first method, all differential equations (linear or nonlinear) are discretized using a numerical integration method and the resulting algebraic equations are solved simultaneously. The solution of the algebraic equations can be nonlinear or linear depending on the equipment that needs to be modeled. This solution approach will be used in this work.

Reference [22] uses a numerical integration substitution method for the network, which embeds the numerical integration formula into the differential equation and rearranges the function into an appropriate form.

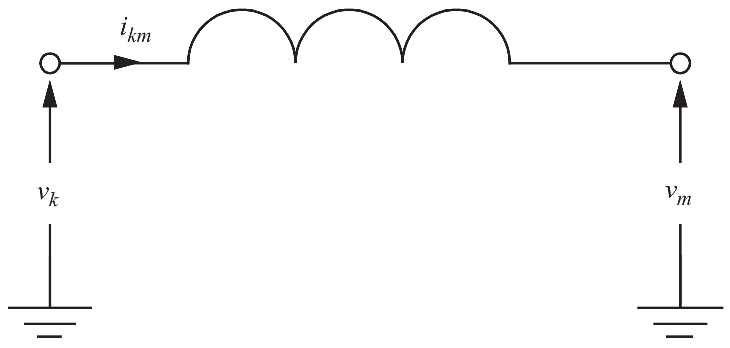


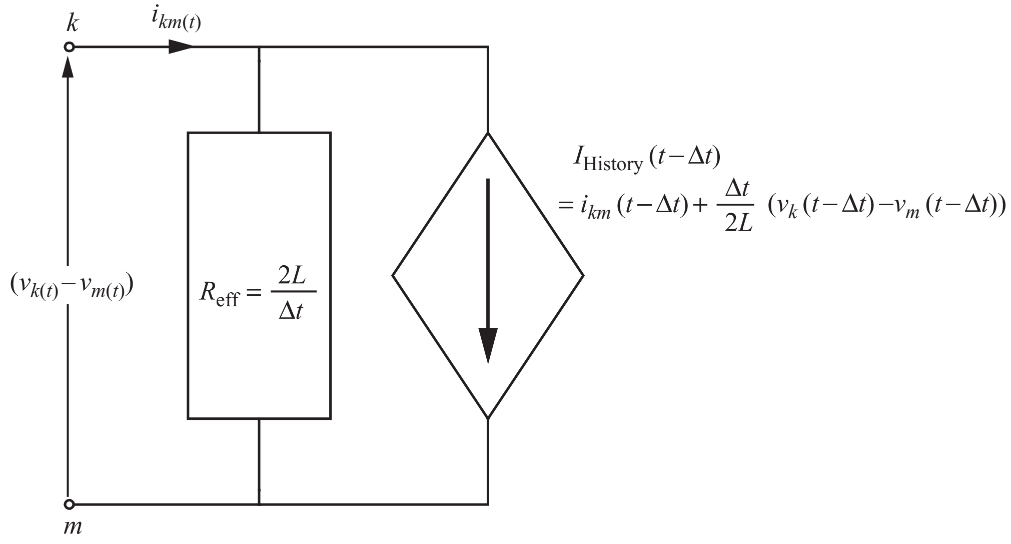
Figure 3.2. Inductor

For example, using a trapezoidal rule the equation for an inductor can be given as

$$\begin{aligned} i_{km}(t) =& i_{km}(t-\Delta t) + \frac{\Delta t}{2L}((v_k - v_m)_{(t)} + (v_k - v_m)_{(t-\Delta t)}) \\ =& I_{HISTORY}(t-\Delta t) + \frac{1}{R_{eff}}(v_k(t) - v_m(t)) \end{aligned} \tag{3.29}$$

In equation 3.29

$$\begin{aligned} I_{HISTORY}(t-\Delta t) &= i_{km}(t-\Delta t) + \frac{\Delta t}{2L}((v_k - v_m)_{(t-\Delta t)}) \\ R_{eff} &= \frac{2L}{\Delta t} \end{aligned}$$



k
$i_{km(t)}$
$I_{\text{History}}(t-\Delta t)$
$= i_{km}(t-\Delta t) + \frac{\Delta t}{2L}(v_k(t-\Delta t) - v_m(t-\Delta t))$
$(v_{k(t)} - v_{m(t)})$
$R_{\text{eff}} = \frac{2L}{\Delta t}$
m

Figure 3.3. Norton equivalent of an inductor

Other elements such as capacitors, resistors, series and parallel RLC branches, or transmission lines can be also represented in a Norton equivalent form using the numerical integration substitution method. For the solution of the equations, all the network components are represented by Norton equivalents and a nodal formulation is used for the solution

$$[G]v(t) = \; i(t) + I_{HISTORY} \tag{3.30}$$

where

- G is the conductance matrix
- $v(t)$ is the vector nodal voltages
- $i(t)$ is the vector of external current sources such as generator current injections or nonlinear load current injections
- $I_{HISTORY}$ is the vector current sources representing past history terms

Equation 3.30 is a linear equation which needs to be solved at each EMT time step. Any nonlinearities such as nonlinear effects in synchronous machines are incorporated by modification of the linear solution method rather than performing non-linear solution of the entire network [77]. The approach for including the nonlinearities in the linear solution process is done by either current source representation (with one time step delay), compensation, or piecewise linear representation.

For more information on these methods the reader is refered to [77].

3.4 EMT simulator benchmarking

An EMT simulator was developed in this research work that can be used for electromagnetic transient simulations. The code for the EMT simulator is written in the C language using the library PETSc. It uses a state variable analysis method as the numerical solution process. The resulting algebraic system after discretization is solved using the Newton method. This method of solution process was chosen for simplicity and to ease the implementation of the implicitly coupled TSEMT simulator.

The proposed EMT simulator was benchmarked against the MATLAB based electromagnetic transients simulation package PowerSystemBlockset [56] on the WECC 9-bus system (Figure B.1). This system has three generators, nine transmission lines and three loads. The loads are modeled as constant impedances and the generators are modeled as GENROU with an IEEE Type 1 exciter on each. As a disturbance scenario, a three phase fault with $R_{fault} = 0.001$ pu was applied on bus 5 at 0.1 seconds and the clearing time of the fault was varied.

Figures 3.4 - 3.6 show the EMT simulator benchmarking results for a stable case i.e. the 9-bus system regains stability following fault clearing. For this test scenario, the three phase fault on bus 5 was cleared at 0.2 seconds, i.e. 6 cycles. Following the three phase fault on bus 5, the generator rotors accelerate due to the difference between the constant mechanical power input and now reduced electrical power output. As a result the generator speeds continue to increase till 0.2 seconds. When the fault gets cleared at 0.2 seconds, generators at buses 1 and 3 start decelerating while generator 2, which is closer to the fault location and has lower inertia than the generator at bus 1, continues to accelerate till 0.8 seconds. Figures 3.5 and 3.6 show the instantaneous three phase voltages for buses 5 and 6. A three phase fault on bus 5 at 0.1 seconds causes its voltages to drop to zero due to the short circuit. Following the fault clearing, these three phase voltages recover and gradually increase to re-attain the pre-fault steady state voltages. Bus 6, which also has a load, experiences depressed voltages following the fault. Eventually, bus 6 voltages also recover to steady state operating values. Although, the results for this stable case scenario are shown only for 1 second, it was verified by running the EMT simulator for 15 seconds that the generator speeds regain synchronous speed and all the bus voltages re-attain pre-fault steady state voltages.

The simulation results for an unstable case are shown in Figures 3.7 - 3.9. For this test scenario, the three phase fault was cleared in 12 cycles, i.e. at 0.3 seconds. As seen in 3.7, generator 2 continues to accelerate even after the fault is cleared and loses synchronism eventually. The other generators also lose synchronism as a result. Generator at bus 1, which has the largest inertia, shows the least speed deviation. The voltages at bus 5 and 6, Figures 3.8 and 3.9, experience oscillations in the voltage as a result of the system instability.

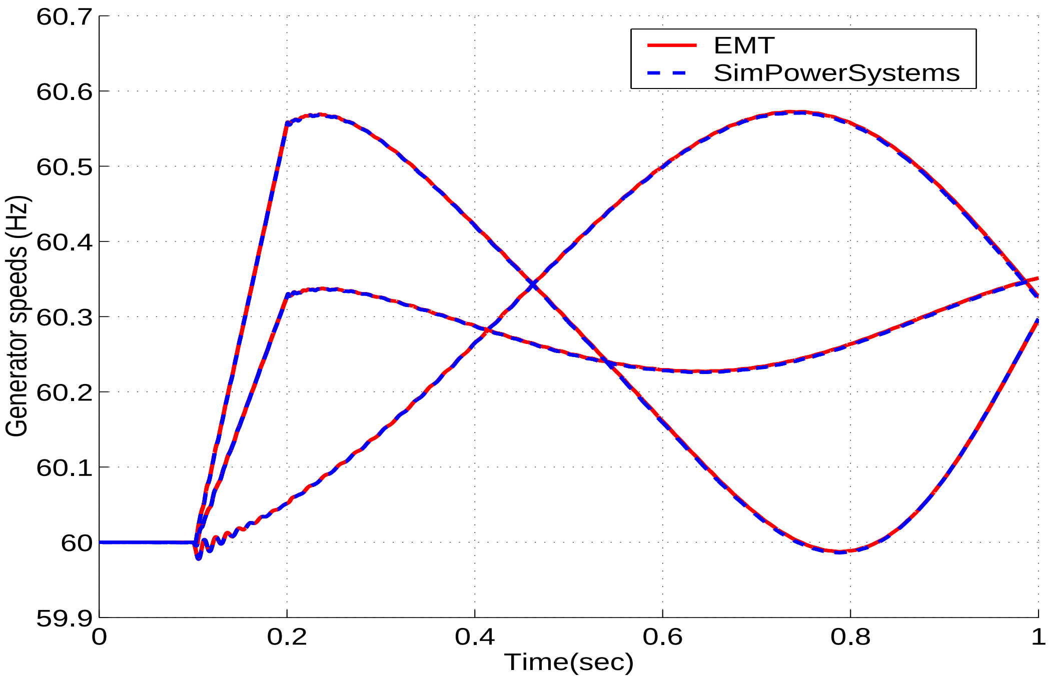


Figure 3.4. Generator speeds for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

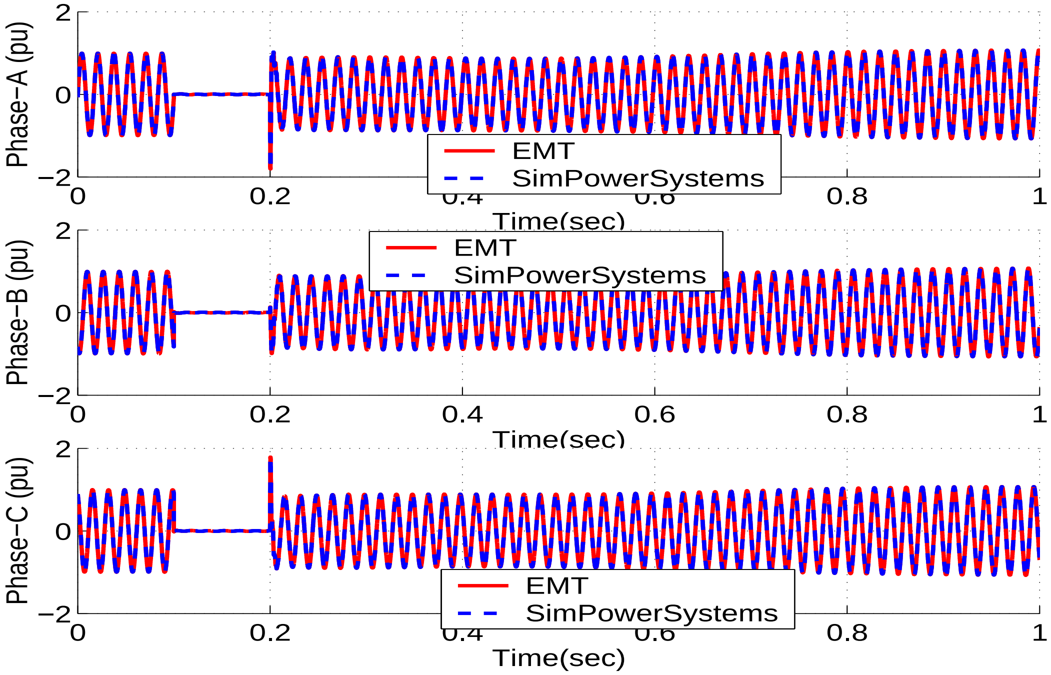


Figure 3.5. Bus 5 instantaneous voltages for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

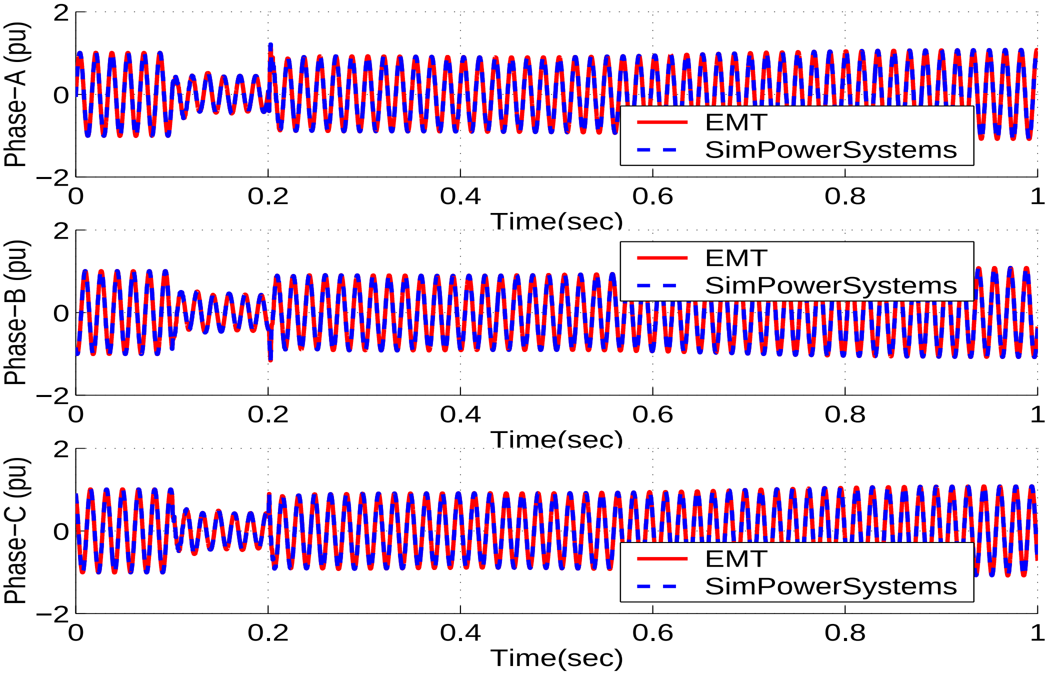


Figure 3.6. Bus 6 instantaneous voltages for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

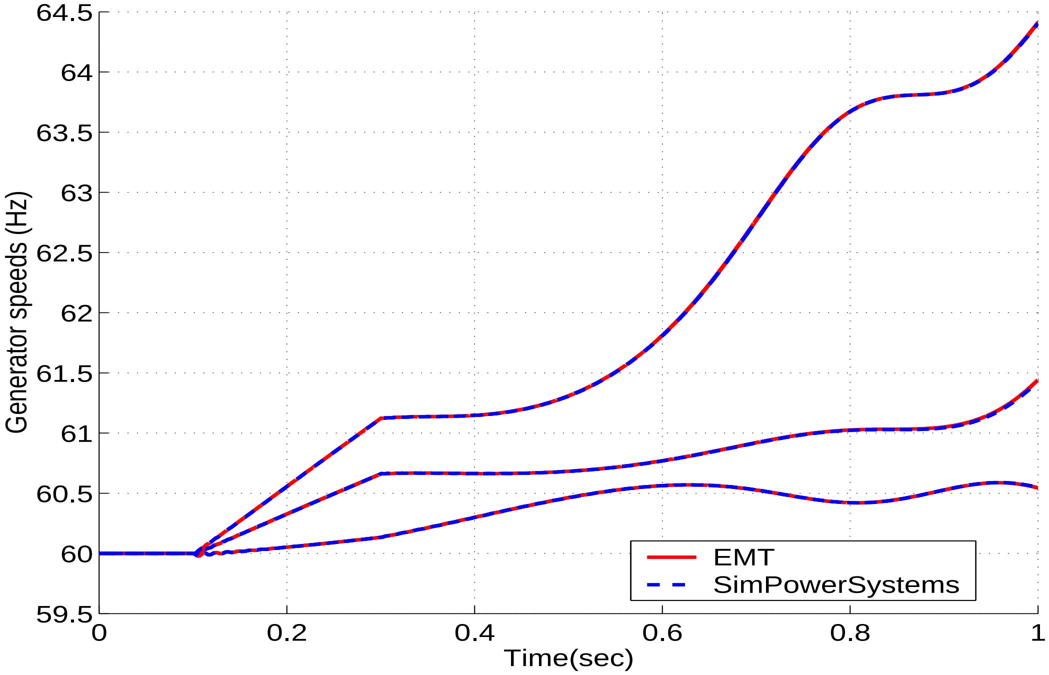


Figure 3.7. Generator speeds for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

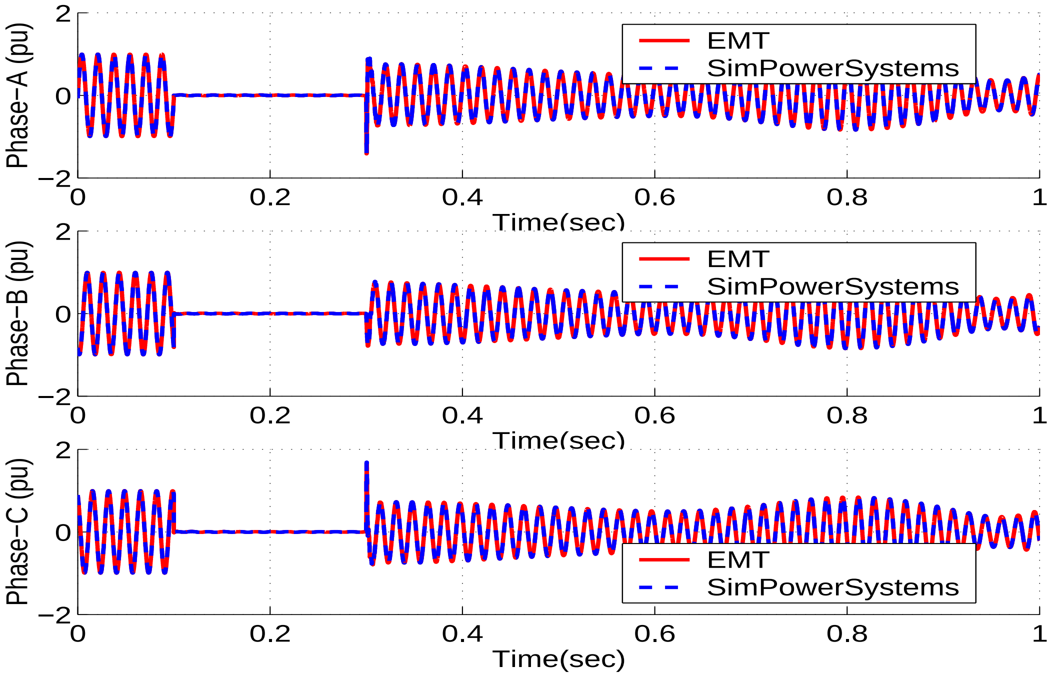


Figure 3.8. Bus 5 instantaneous voltages for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

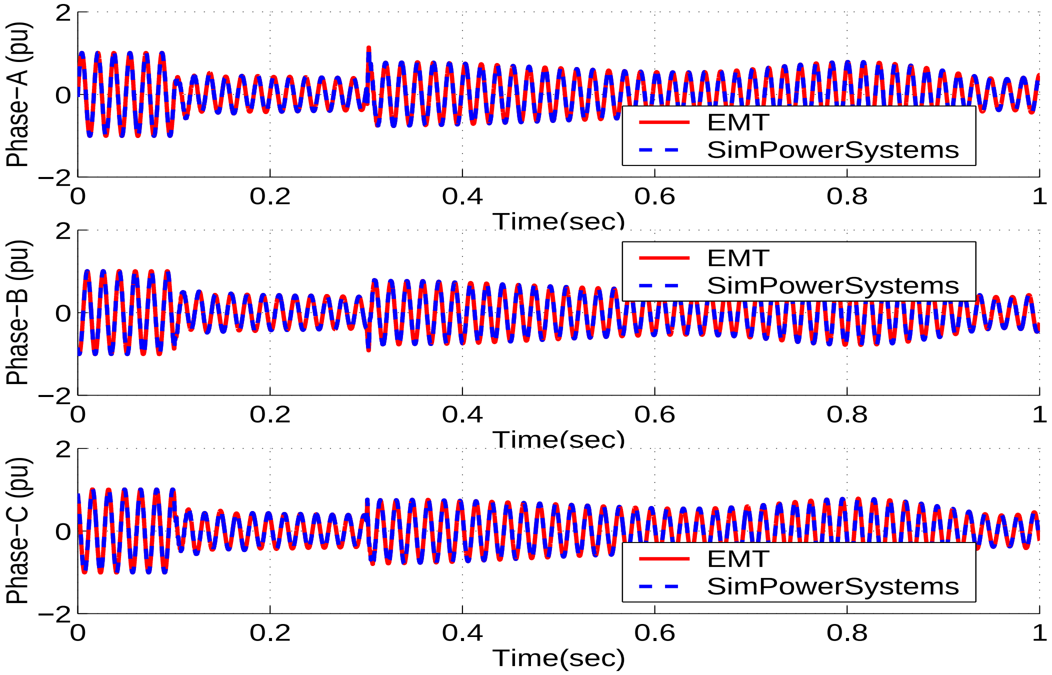


Figure 3.9. Bus 6 instantaneous voltages for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

CHAPTER 4

HYBRID SIMULATORS

A hybrid simulator connects a transient stability simulator and an electromagnetic transient simulator running separately and connects these two simulators, running separately at different time steps with an interface or sequence of actions to exchange data as well as reduced circuitry. The need for the interface protocol and the associated circuitry is due to the differences in TS and EMT as shown in Table 4.1.

Table 4.1. Differences between TS and EMT

Property	TS	EMT
Time step	On the order of milliseconds	On the order of microseconds
Network Modeling	Balanced positive sequence	Three phase unbalanced
Voltages and currents	Phasor	Instantaneous

The idea of hybrid simulation was first proposed by Hefernan, et. al., in [28] to simulate combined HVAC-HVDC systems. They modeled a HVDC link in detail within a stability based AC system framework, thus exploiting the advantages of both EMT and TS. They achieved this by executing TS and EMT alternately with periodic coordination of the results. Reference [58] proposed that the boundary of the interface should be extended into the AC network further for taking into consideration the effect of harmonics generated by power electronics on the AC network.

Reference [6] presented another approach to take the harmonics into account. In the EMT program, the network portion simulated by the TS program is represented by a frequency-dependent equivalent, instead of a simple fundamental frequency equivalent circuit used by Heffernan and Reeve. Reference [71] basically adopted the approaches described above, i.e., extending the interface location into the AC network to some extent, and at the same time, representing the network simulated by the TS program with a frequency-dependent equivalent. Also, Kasztenny, et. al., [42] have discussed a general method for linking different modeling techniques such as waveform-type, phasor-type, and algebraic-type simulation techniques into one complete model.

In the hybrid simulator, the power system network is partitioned into two sub networks; a large network (TS domain of operation) and a smaller network run with EMT. The large network has been called external system[69], [58], [6], electromechanical transient network[70], TS-program subsystem[24], while the smaller system has been called detailed system[69], [58], [6], EMT network[70], instantaneous network[74]. In this paper, the larger network will be called the external system and the smaller system will be called the detailed system. To connect the external system simulated with TS and the detailed system simulated with EMT, an interface is required over space, time and waveform. These three interfaces and the related terminology are explained in the next section.

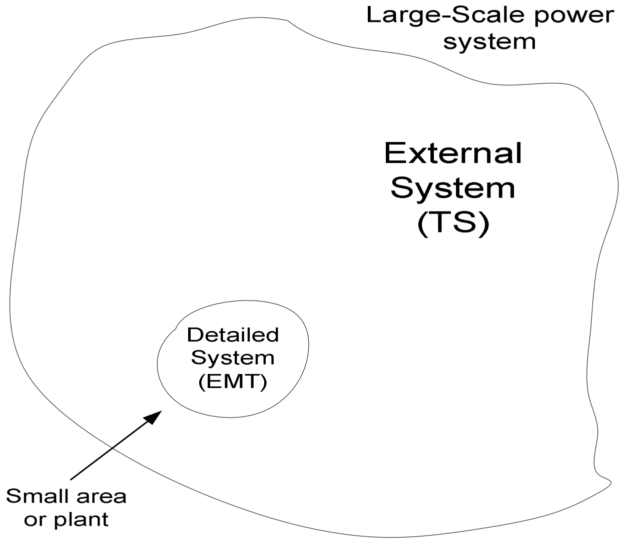


Figure 4.1. Detailed and external system

4.1 Interface/Boundary buses

Buses through which the detailed and the external system interact and exchange information are called the boundary buses. There is no single method for selecting the detailed system. It depends on the area of the power system that needs to be studied in detail, as well as the and requires device-level modeling. The location of the interface buses depends on the waveform distortion and phase imbalance at the interface buses. The first work on a hybrid simulator used the location of the converter bus as the detailed system boundary[28] and the interface location was chosen as the converter bus. The choice of the converter bus as the interface bus location was done to keep the detailed system to a minimum. It was shown later that waveform distortion and phase imbalance affect the accuracy of the hybrid simulator. Hence, [58] used a strategy of extending the detailed system to incorporate more of the external system. However, expansion of the detailed system increases the complexity of the detailed system modeling, and the number of interface buses required. Consequently, the expansion diminishes the efficiency of the hybrid simulator[6]. In[6], it was shown that the detailed system can be kept at a minimum by using a frequency dependent equivalent of the external system.

4.2 Network equivalents

The development of network equivalents is a very well-explored topic. By replacing a huge network with smaller set of circuits, huge computation resources can be saved while retaining the required computational accuracy. Different types of simulators have different requirements for network equivalents. For electromagnetic transient simulators, an equivalent must reflect the dynamics of the original network over a wide frequency range. The equivalent of the external system for an EMT simulator has been done in various ways such as fundamental frequency Thevenin equivalent, fundamental frequency Norton equivalent, frequency dependent Norton[58] and

Thevenin equivalents[6]. Multi-frequency dependent network equivalents of the external system also have been also proposed in the literature[71],[69],[6]. Several forms of equivalents of the detailed system for TS have been proposed in the literature. These include PQ load[71],[58],[68], Norton/Thevenin equivalents[58],[24], equivalent impedance[58],[67], and current injection[28],[71].

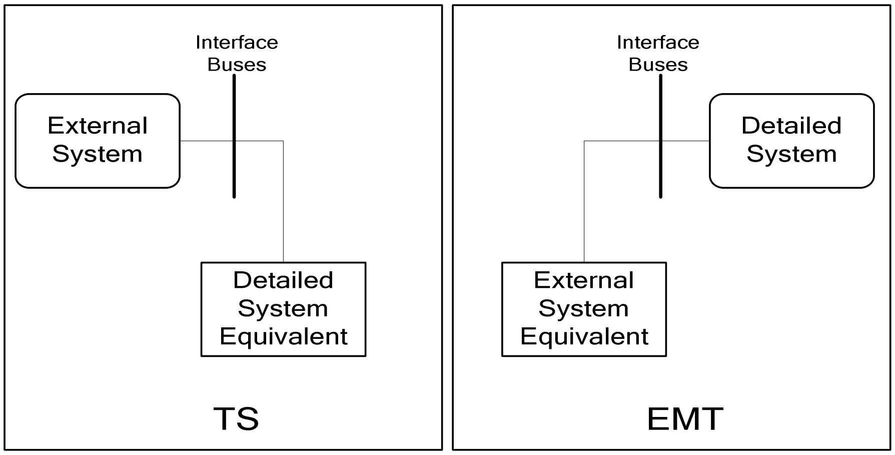


Figure 4.2. Equivalents for the hybrid simulator

4.3 Interaction protocol

Since the TS and EMT run at different time steps, synchronization of these simulators is required for data exchange. This synchronization is done through predefined sequential actions which coordinate the data exchange between TS and EMT simulators[35]. Both serial and parallel interaction protocols have been proposed so far. In serial protocols, only one simulator either TS or EMT, runs while the other is idle. In parallel protocol, both simulators run at the same time.

4.4 Data conversion

TS uses fundamental frequency phasors for voltages and currents while EMT uses instantaneous voltages and currents. Thus there is a need for conversion from phasor to instantaneous and vice versa for exchanging data. The phasor-to-instantaneous

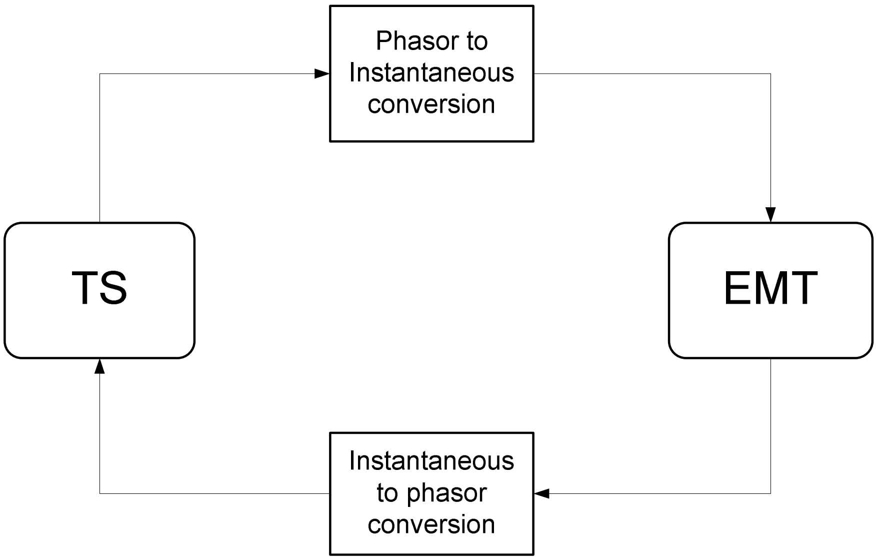

Phasor to Instantaneous conversion
TS
EMT
Instantaneous to phasor conversion

Figure 4.3. TS and EMT waveform interface

conversion is done though a signal generator (typically sine wave) controlled by amplitude, phase and frequency[42]. The conversion from instantaneous to phasor could be done either using the Fast Fourier transform[74],[42] or a curve fitting technique[6],[66].

4.5 Existing interaction protocols

A literature survey on the existing hybrid simulators reveals that the existing interaction protocols are divided into two classes: serial and parallel. The definition of the terms serial and parallel here is strictly in the sense of how TS and EMT interact and does not mean serial and parallel as used in the computing world. Before delving into the details of the different interaction protocols a few symbol definitions are for the following discussion:

Δt_{TS} TS time step

Δt_{EMT} EMT time step

k Number of time steps for EMT to synchronize with TS

X_{TS}	TS dynamic variables
x_{EMT}	EMT dynamic variables
Y_{TS}	TS algebraic variables
y_{emt}	EMT algebraic variables

In addition, the following terms are restated for clarity:

External system	TS network
Detailed system	EMT network
External system equivalent	equivalent of TS network passed to EMT.
Detailed system equivalent	equivalent of EMT network passed to TS.

4.5.1 Serial Interaction Protocol. Serial interaction protocols were the first interaction protocols to be developed. The main aim of the serial interaction protocol is to interface seperate TS and EMT software programs running at different time steps. A natural choice made for the serial interaction was to allow the TS and EMT simulators to run independently and exchange data at regular intervals. The exchange of data was done every TS time step. Since serial interaction protocols mainly serve as an interface, their drawback is that both TS and EMT cannot be run simultaneously, i.e., TS is idle when EMT is running and vice-versa. For example if the hybrid simulator was to run on two processors with processor 0 running TS and processor 1 running EMT, then processor 1 would have to wait until TS was finished on processor 0 and vice-versa. Hence they cannot be used for real-time applications. In the serial protocol, TS and EMT run sequentially exchanging data at each TS time step. A single TS time step for a for hybrid simulator using serial interaction protocol is shown in Figure 4.4.

The sequence of actions taken in a serial interaction protocol are as follows:

1. TS passes the external system equivalent to EMT at time t.

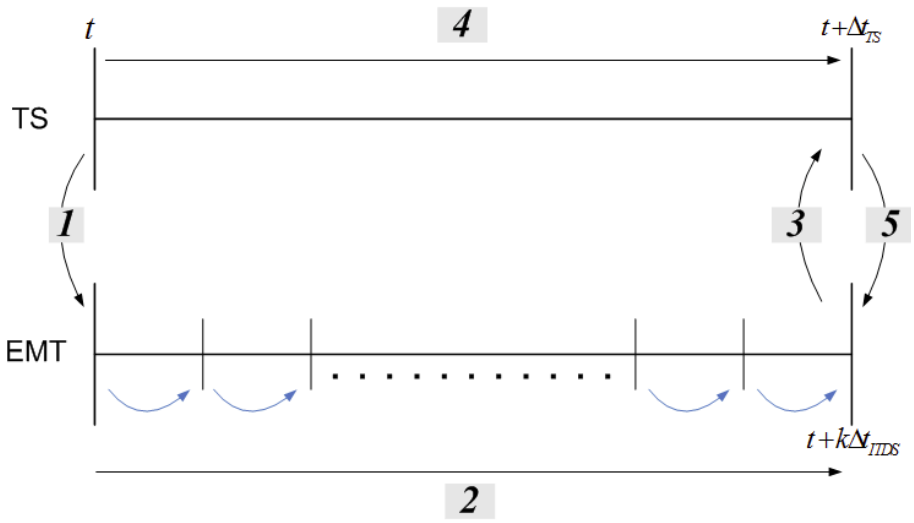


Figure 4.4. Serial interaction protocol for one TS time step

2. EMT solves the detailed system equations for $x_{EMT}(t+\Delta t_{EMT}), y_{EMT}(t+\Delta t_{EMT})$. Once it computes the solution, it proceeds to compute the solution for $x_{EMT}(t+2\Delta t_{EMT})$ and so on, till it computes the solution at time $t+k\Delta t_{EMT}$ which is equal to $t+\Delta t_{TS}$.

3. At this point, EMT computes the equivalent of the detailed system and passes it to TS.

4. The TS simulator, which is still at time t, solves the external system for the next TS time step solution $X_{TS}(t+\Delta t_{TS}), Y_{TS}(t+\Delta t_{TS})$.

5. TS passes the external system equivalent at time $t+\Delta t_{TS}$.

This completes one time step of the hybrid simulator and 1-4 is repeated for future time steps. Note that the external system equivalent passed to EMT is either constant for the entire TS time step or extrapolated from some history data which can introduce errors.

4.5.2 Parallel Interaction Protocol. Parallel interaction protocols [69],[66] have been proposed to remedy the drawback in serial interaction protocols when the two simulators cannot run simultaneously. Simultaneous execution of TS and EMT is critical for real-time simulation applications and hence parallel interaction protocols were proposed. In the parallel interaction protocol, TS and EMT simulators are never idle. The basis of this protocol is that after every n, where $n < k$ EMT time steps, EMT passes the detailed system equivalent for each TS iteration. If each TS iteration requires the same amount of time as running n EMT time steps, then the two simulators run simultaneously.

The working of a parallel interaction protocol from [69] is described in the following discussion and its interaction diagram is shown in the figure 4.5.

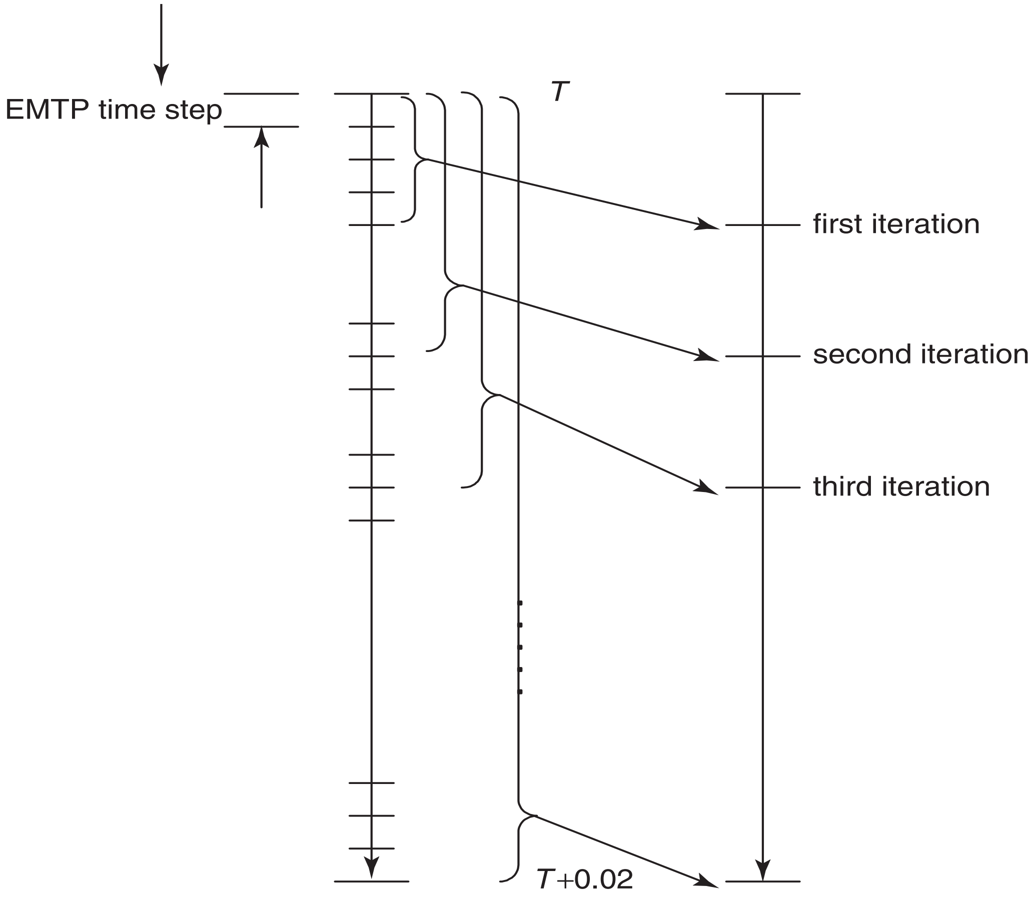


Figure 4.5. Parallel interaction protocol for one TS time step [69]

The parallel interaction protocol works as follows

1. TS passes the external system equivalent to EMT at time T and waits for the detailed system equivalent from EMT.
2. EMT starts running and after n time steps, it passes the detailed system equivalent to TS.
3. At this point TS starts its first iteration while EMT continues marching along in time.
4. The two simulators run concurrently where the updated detailed system equivalent is passed to TS at every iteration.
5. At the end of each TS time step, TS updates the external system equivalent and passes it to EMT. (not shown in the diagram)

Again note that the external system equivalent used by EMT is either constant or extrapolated based on some history data and only updated at each TS time step.

4.6 Existing combined electromechanical and electromagnetic transients simulation strategies

Using the interaction protocols, the combined electromechanical-electromagnetic transients simulation strategy can be divided into two categories; one for studying fast dynamics locally and the other for studying the slow dynamics over the entire power system [35]. The simulation strategy shown in Figure 4.6 is used for studying faster dynamics of the local area in detail and is only run for a short simulation period. This simulation strategy works as follows:

1. The network is split, before the simulation starts, into a detailed system and an external system.

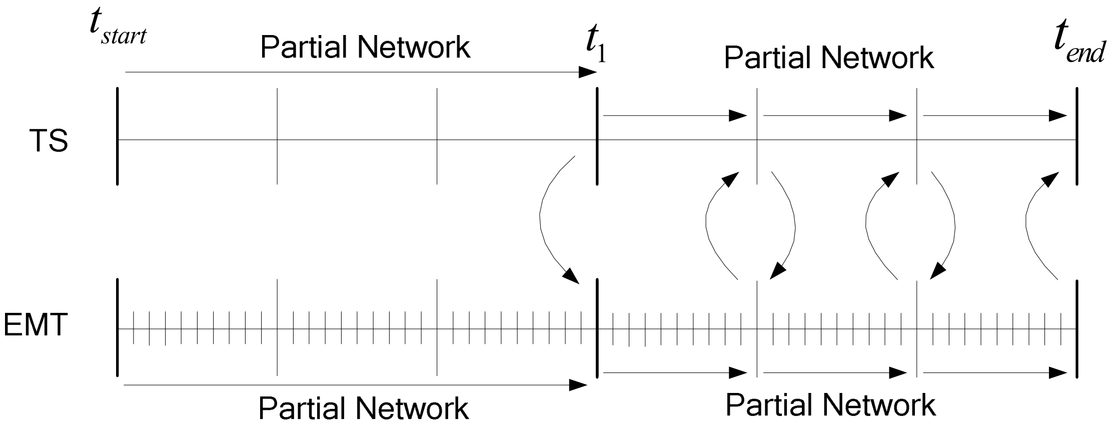


Figure 4.6. Hybrid simulation strategy for studying short-term dynamics

2. TS and EMT run independently till time t_1. There is no data exchange assuming that the hybrid simulation starts in steady state.

3. At time t_1, a disturbance is applied and the hybrid simulation begins. The equivalent of the external system is passed to EMT at time t_1.

4. TS and EMT run serially exchanging data at each TS time step according to the above interaction protocol until end time.

The hybrid simulation strategy for studying the system-wide long term dynamics is shown in figure 4.7.

1. TS is run initially on the complete network in the pre-disturbance period.

2. At time t_1, a disturbance is applied and the hybrid simulation begins. The system is split into an external system and a detailed system and equivalent of the external system is passed to EMT.

3. The hybrid simulation continues until time t_2, where it is assumed that the faster dynamics in the detailed system have died down and the entire system

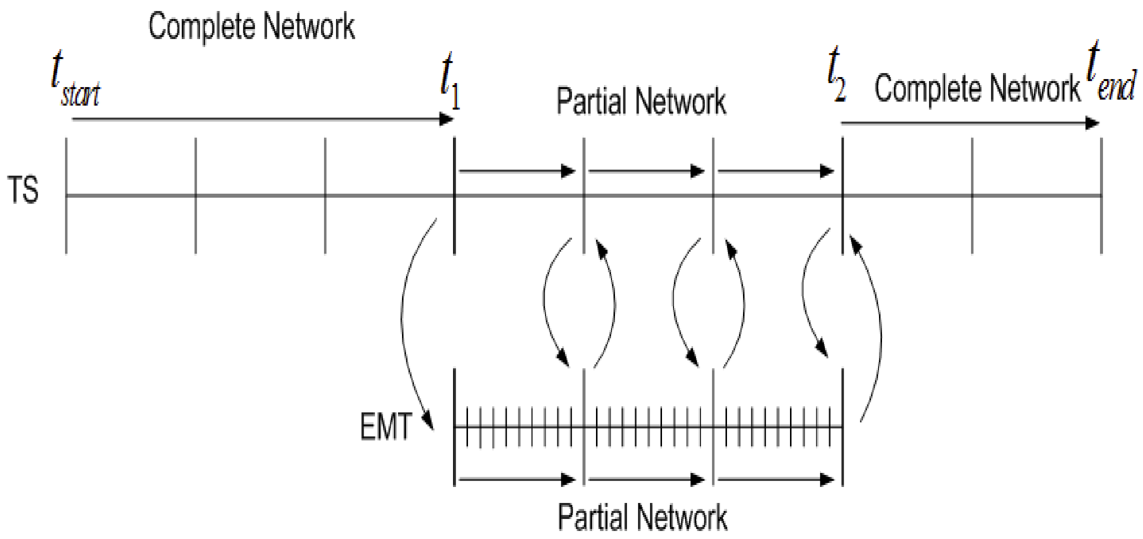


Figure 4.7. Hybrid simulation strategy for studying long-term dynamics

can be merged again. Such a decision of merging can be based on choosing a pre-defined time or checking the boundary bus voltages and currents.

4. The system is merged back at time t_2 and the TS simulator is run on the entire network until t_{end}.

CHAPTER 5
PROPOSED THREE PHASE TRANSIENT STABILITY SIMULATOR (TS3PH)

5.1 Motivation

Dynamic security assessment (DSA) of large-scale electrical power systems is done by existing transient stability simulators that use a balanced per phase model of the transmission network. Due to the balanced per phase assumption, unbalanced phenomena due to large single phase loads, unbalanced disturbances, untransposed transmission lines, single phase feeders, single phase switching operations cannot be studied realistically.

In recent years, power systems have grown in size and complexity due to network expansion and addition of more components to meet the ever-increasing demand. The deregulation of the power industry is motivating economic operation and thus operating the electrical network to its fullest potential and hence near its stability limits. Under such circumstances the analysis of single phase disturbances becomes even more important.

A balanced transmission network model was used in the first developed transient stability analysis tools to make the computational problem tenable. Over the last decade there has been tremendous development in computing architectures enabling faster processor speed, bigger memory, and the ability to solve the problems using multiple processors. Moreover, there has been a lot of research on speeding up the solution of linear systems.

We propose a three phase transient stability simulator which uses a full three phase network model of the transmission network. The ability to model all three phases can provide a more realistic picture of the dynamics of the individual three phases. When power systems operate at their limits, the margin between stability and instability is small and even single phase disturbances could trigger dynamic instability. Hence, such single phase disturbances need to be analyzed and a three phase transient stability simulator can provide details of the individual phase dynamics. Moreover, due to modeling of all the three phases, relay operations on individual phases can be analyzed and the relay settings for single phase tripping can be better ascertained.

Load modeling is one of the most critical components in studying dynamics and there is a lot of ongoing research. For transient stability analysis, the load model at any bus is assumed to be the same for all the three phases. A three phase transient stability simulator has the capability to model all the individual phase loads by the same or different load models and thus provide a better understanding of how the load dynamics on each phase affect the system stability.

For the rest of this thesis, the three phase transient stability simulator will be referred to as TS3ph. This chapter details the equipment modeling and the numerical solution scheme used in the newly developed three phase transient stability simulator. It presents benchmarking of the three phase transient stability simulator against a commerical positive sequence TS simulator PSS/E. TS3ph code organization and implementation are provided and the results of experimentation with various reordering strategies and linear solution schemes is presented.

5.2 Equipment modeling

We extend the modeling of different power system equipment from a balanced per phase model used in TS to three phase modeling for TS3ph in this section.

5.2.1 Generator subsystem modeling. The generator subsystem modeling used for TS3ph is similar to that described in chapter 2. The generator and exciter models follow the same formulation since they are expressed on the machine dq axis reference frame. The difference between TS and TS3ph modeling for generators is in the machine to network transformation. While TS does a conversion from positive sequence to dq TS3ph does a conversion from abc to dq.

5.2.2 Machine to network transformation. The machine-network transformation for TS3ph does the conversion of three phase phasor voltages $\bar{V}_a, \bar{V}_b, \bar{V}_c$ to V_d, V_q and the machine stator currents I_d, I_q to three phase phasor currents $\bar{I}_a, \bar{I}_b, \bar{I}_c$. The conversion equations in rectangular form and with $\hat{\delta} = [\delta, \delta - 2\pi/3, \delta + 2\pi/3]$ are

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \frac{1}{3} \begin{bmatrix} \sin\,\hat{\delta} & -\cos\,\hat{\delta} \\ \cos\,\hat{\delta} & \sin\,\hat{\delta} \end{bmatrix} \begin{bmatrix} V_{genD,a} \\ V_{genD,b} \\ V_{genD,c} \\ V_{genQ,a} \\ V_{genQ,b} \\ V_{genQ,c} \end{bmatrix} \tag{5.1}$$

$$\begin{bmatrix} I_{genD,a} \\ I_{genD,b} \\ I_{genD,c} \\ I_{genQ,a} \\ I_{genQ,b} \\ I_{genQ,c} \end{bmatrix} = \begin{bmatrix} \sin\,\hat{\delta} & \cos\,\hat{\delta} \\ -\cos\,\hat{\delta} & \sin\,\hat{\delta} \end{bmatrix} \begin{bmatrix} I_d \\ I_d \end{bmatrix} \tag{5.2}$$

5.2.3 Network subsystem modeling. In TS3ph we use a full three phase model of the transmission network. The relationship between the three phase phasor currents and voltages is as follows:

$$\bar{Y}_{bus3ph}\bar{V_{abc}} = \bar{I}_{inj,abc} \tag{5.3}$$

Here, $\bar{Y}_{bus3ph}$ is the three phase complex “Y” matrix. $\bar{I}_{inj,abc}$ is vector of the sum of complex generator current $\bar{I}_{gen,abc}$ and load current $\bar{I}_{load,inj}$ injected at the load buses.

Splitting the complex three phase voltage vector $\bar{V}_{abc}$ into real and imaginary parts $V_{D,abc}$ and $V_{Q,abc}$, equation 5.3 in current balance form is

$$\begin{bmatrix} G_{3ph} & -B_{3ph} \\ \\ B_{3ph} & G_{3ph} \end{bmatrix} \begin{bmatrix} V_{D,abc} \\ \\ V_{Q,abc} \end{bmatrix} = \begin{bmatrix} I_{Dinj,abc} \\ \\ I_{Qinj,abc} \end{bmatrix} \tag{5.4}$$

In equation 5.4, G_{3ph} and B_{3ph} are the real and imaginary parts of the complex three phase $\bar{Y}_{bus}$ matrix. The three phase complex $\bar{Y}_{bus}$ matrix can be built in a similar way as the per phase Y matrix.

5.2.4 Load subsystem modeling.

The modeling of load done for TS3ph follows that of single phase TS with the exception that now individual loads can be put on each phase. Currently the implemented loads in TS3ph assume that the load models are the same on all the three phases. The impact of different static or dynamic load models on each phase is a future research topic.

5.3 Equations and variables

The equations for the proposed three phase transient stability simulator with the network equations written in current balance form and the three phase network voltages and currents in rectangular form are as follows:

$$\frac{dx_{gen}}{dt} = f(x_{gen}, I_{dq}, V_{DQ,abc}) \tag{5.5}$$

$$0 = h(x_{gen}, I_{dq}, V_{DQ,abc}) \tag{5.6}$$

$$\begin{bmatrix} G_{3ph} & -B_{3ph} \\ B_{3ph} & G_{3ph} \end{bmatrix} \begin{bmatrix} V_{D,abc} \\ V_{Q,abc} \end{bmatrix} = \begin{bmatrix} I_{genD,abc}(x_{gen}, I_{dq}) \\ I_{genQ,abc}(x_{gen}, I_{dq}) \end{bmatrix} - \begin{bmatrix} I_{loadD,abc}(x_{load}, V_{DQ,abc}) \\ I_{loadQ,abc}(x_{load}, V_{DQ,abc}) \end{bmatrix} \tag{5.7}$$

$$\frac{dx_{load}}{dt} = f_2(x_{load}, V_{DQ,abc}) \tag{5.8}$$

Grouping all the dynamic variables together in one set and all the algebraic variables in another set, the TS3ph equations can be described by the differential-algebraic model:

$$\begin{aligned} \frac{dx}{dt} &= f(x,y) \\ 0 &= g(x,y) \end{aligned} \tag{5.9}$$

where

$$\begin{aligned} x &\equiv [x_{gen}, x_{load}]^t \\ y &\equiv [I_d, I_q, V_{D,abc}, V_{Q,abc}]^t \end{aligned}$$

x_{gen} are the dynamic variables for the generator subsystem, i.e., the generator and exciter dynamic variables. x_{gen} for each generator varies depending on the types of generator, exciter, and other control equipment models. For a generator modeled using GENROU model and an IEEET1 exciter model, xgen would be as follows:

$$x_{gen} \equiv \left[E_q^{'}, E_d^{'}, \psi_{1d}, \psi_{2q}, \delta, n, E_{fd}, R_F, V_R\right]^t$$

The number of variables in x_{load} equals the number of dynamic load variables. If there are no dynamic load variables then the size of the x_{load} vector is 0. For an induction motor model, the dynamic variables are $x_{load} = \left[e_q^{'}, e_d^{'}, s\right]^t$

If the voltage variables are arranged as $[V_{Dai}, V_{Dbi}, V_{Dci}, V_{Qai}, V_{Qbi}, V_{Qci}]^t$, where $i = 1...\text{nbus}$, then using this ordering, the "Y" matrix in 5.4 becomes a matrix with a 6 X 6 block for each branch connection and for each bus on the diagonal.

5.4 Discretization and Numerical solution

TS3ph uses a fixed step implicit trapezoidal scheme for discretization as it is easy to implement and has numerical A-stablility properties. A literature survey on the transient stability simulators also revealed that the implicit trapezoidal scheme is the prefered discretization scheme.

Using the implicit trapezoidal scheme the equations to be solved by TS3ph are

$$\begin{gathered}x(t+\Delta t) - x(t) - \frac{\Delta t}{2}\left(f(x(t+\Delta t), y(t+\Delta t)) + f(x(t), y(t))\right) = 0 \\ g(x(t+\Delta t), y(t+\Delta t))) = 0\end{gathered} \tag{5.10}$$

Equation 5.10 is then solved iteratively using Newton's method at each time step. The linear system to be solved in each Newton iteration is

$$\begin{bmatrix} J_{xx} & J_{xy} \\ J_{yx} & J_{yy} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = - \begin{bmatrix} F_x \\ F_y \end{bmatrix} \tag{5.11}$$

5.5 TS3ph implementation details

The code for the proposed three phase transient stability simulator, TS3ph, was written in the C language using the library PETSc. The current TS3ph implementation uses the development version of PETSc. The details of PETSc library are given in Appendix A and the parallel implementation details are deferred to Chapter 8.

5.5.1 List of models implemented. Table 5.1 lists available models implemented for TS3ph.

GENROU is a 6th order round-rotor generator model in PSS/E. The saturation effects in this model have been neglected. The two additional equations for this model arise from algebraic equations for stator currents I_d and I_q. GRDC is a reduced version of the GENROU model and is described in [44]. GRDC is a 4th order generator model

Table 5.1. Equipment models in TS3ph

Equipment	Model Name/Type	Number of variables
Generator	GENROU	8
Generator	GRDC[44]	6
Exciter	IEEET1	3
Turbine-Governor	GAST	3
Transmission line	π	0
Network bus	BUS	6
Constant impedance load	CZ	0
Induction motor model	CIM5	3
Voltage dependent impedance	VDIL	0
Fault	FAULT	6

with stator dynamics, damper winding fluxes and saturation neglected. IEEET1 is the IEEE type 1 exciter model and GAST is the steam turbine-governor model both described in the PSS/E manual [55].

The transmission network model used in this research work for TS is a three phase coupled π transmission network model. There are three types of load models currently implemented viz., constant impedance load model, induction motor model with one rotor cage, and voltage dependent impedance load model [**?**]. A three-phase fault model is also available which can model single-phase, line-line, and three phase faults.

5.5.2 Steady state initialization and data preparation. Typically transient stability simulations begin in steady state and a disturbance is applied at a later point in time. To obtain the initial steady state operating point, a steady state power flow is executed for determining the initial bus voltages. This is followed by initialization of the generator dynamic variables corresponding to the steady state operating point.

The power flow and the generator dynamic variable initialization are not currently implemented in the TS3ph code. Instead, these steps are done via a code written in Matlab. This Matlab code does the following:

- Run the steady state balanced per phase power flow using the Matlab based package MATPOWER [79].
- Initialize the generator dynamic variables to steady state values.
- Assemble the three phase admittance matrix and save it to a binary file.
- Save generator, load, and network data to ascii files.
- Save the adjacency graph and vertex weights to ascii files (required for parallel runs).
- Save the network steady state voltages and the generator initial dynamic variables to an ascii file.

The TS3ph code reads the different ascii files and the binary file containing the three phase admittance matrix and sets up the different data structures.

Reading of power flow and dynamic data from standard data files, adding a power flow solver to TS3ph, and dynamic variable initialization is a part of the future work.

5.5.3 Code organization. An electrical power system can be thought of as divided into three major subsystems; generator, network, and load. Accordingly, the TS3ph code has been organized according to this subsystem division, with each subsystem having its own top-level 'subsystem' data structure having information about its subsystem models. These top-level subsystem data structures are then packed in the root 'system' data structure. This subsystem hierarchical structure was very helpful for developing, maintaining, and extending the code.

The solution X and the residual vector F also follow this hierarchy. At every Newton iteration, the system solution vector is split into the individual subsystem vectors and the function evaluation for each subsystem is done.The residual vector of each subsystem is then merged in the single system residual vector. These splitting and gathering operations do not cost extra memory since they are merely address mapping operations.

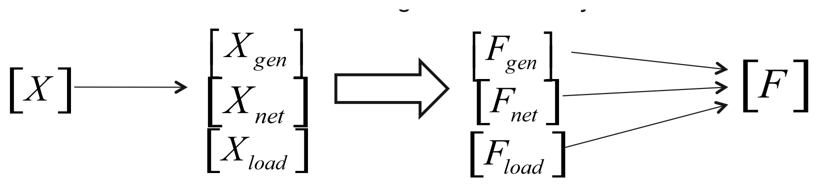


Figure 5.1. Packing and unpacking of solution and residual vector

5.6 Simulation results

The TS3ph simulator was benchmarked on the WECC 9-bus system with the commercial positive sequence TS simulator PSS/E version 30 on a Windows Vista PC. The benchmarking scenario was similar to that of the EMT simulator, i.e., a fault is applied on bus 5 and the clearing time of the fault is varied to analyze the stability of the system. The generators at buses 1,2, and 3 are modeled as GENROU with an IEEET1 exciter model and all the loads are modeled as constant impedance loads.

The benchmarking simulations were run for 3 seconds and the step size used for numerical integration was 1 cycle, i.e., 0.01667 seconds.

5.6.1 Three phase faults. For the first benchmarking scenario, a three phase balanced fault was applied on bus 5 at 0.1 seconds and removed at 0.2 seconds. Figures 5.2-5.5 show the comparison of the TS3ph simulator with PSS/E for this test case. As seen in figure 5.2, the generator rotors accelerate following the fault at 0.1

seconds and continue to accelerate until 0.2 seconds. When the fault gets cleared at 0.2 seconds, the generators at buses 1 and 3 start decelerating while the generator at bus 2 continues to accelerate till 0.8 seconds. After 0.8 seconds, the generator at bus 2 also starts decelerating and eventually regains synchronous speed. Figure 5.3 shows the dynamics of the exciter field voltages for the three generators. Following the fault, the exciter field voltages increase to provide more field current to the generator for controlling the terminal voltage.

Figure 5.4 show the three phase voltages for bus 5 obtained from TS3ph. Since the fault is balanced the phasor voltages for all the three phases are identical. This individual phase information is not available in PSS/E since it uses a positive sequence representation for voltages. Figure 5.5 shows the comparison of bus 5 positive sequence voltage, obtained directly from PSS/E, and computed for TS3ph.

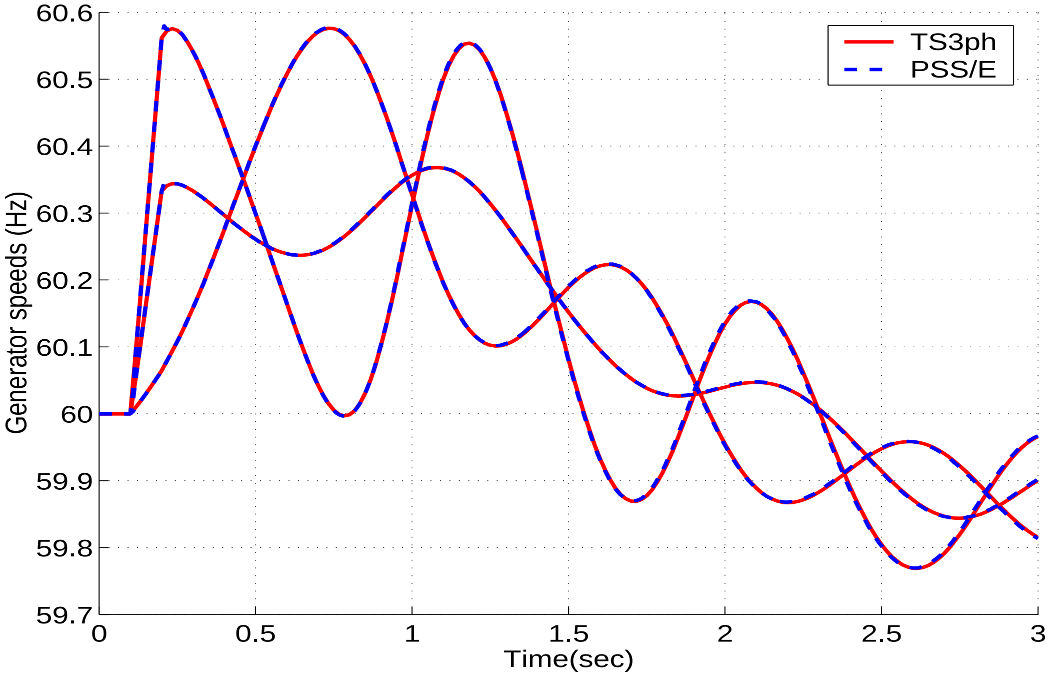


Figure 5.2. Generator speeds for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

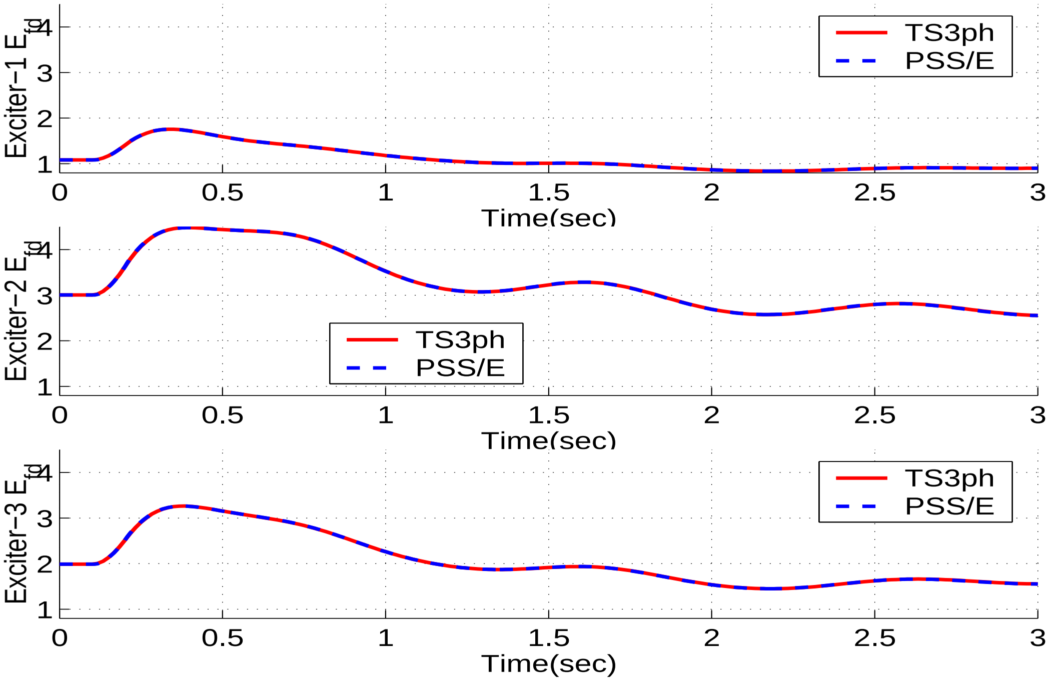


Figure 5.3. Exciter voltages for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

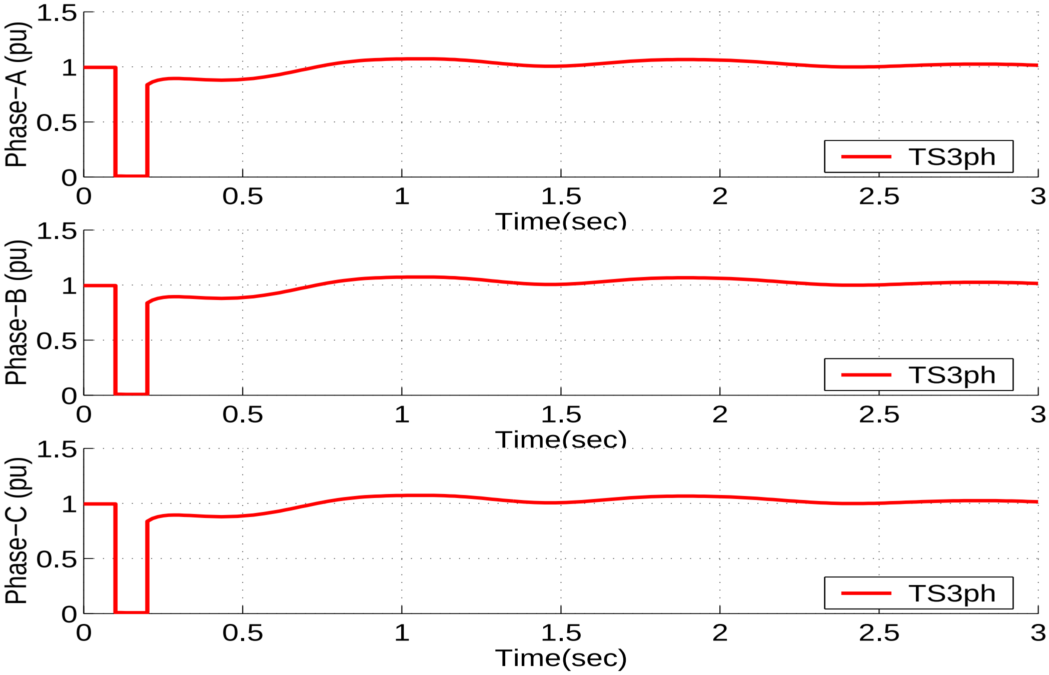


Figure 5.4. Bus 5 three phase phasor voltages for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

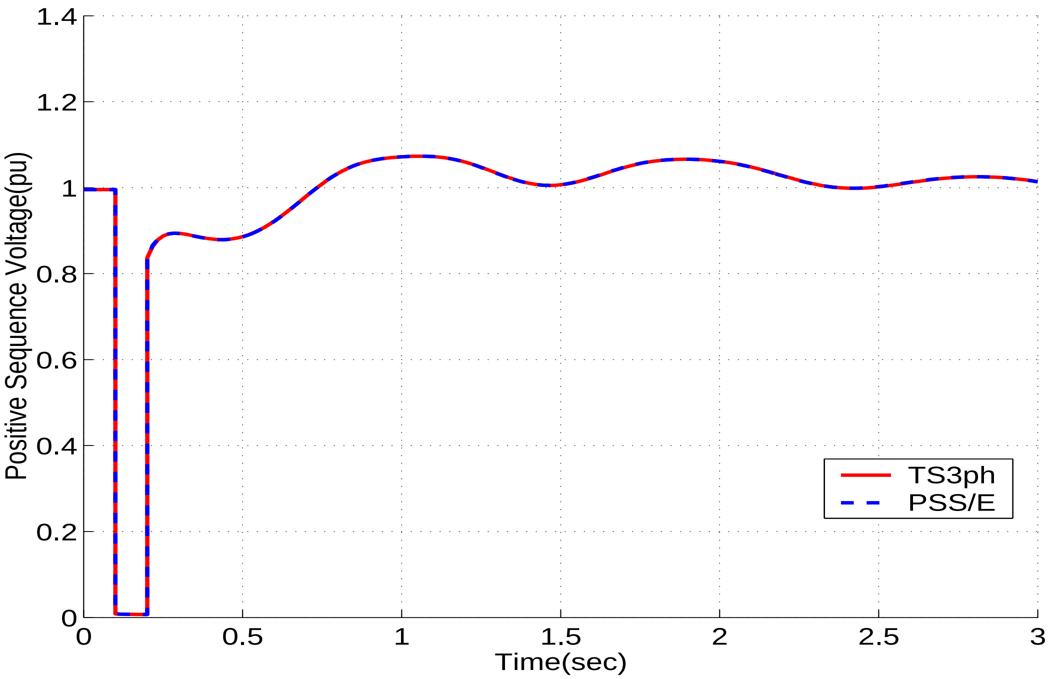


Figure 5.5. Bus 5 positive sequence voltage for a three phase fault on bus 5 from 0.1 sec to 0.2 sec

Next, the fault was extended to 12 cycles, until 0.3 seconds on bus 5. As seen, from the results shown in Figures 5.6-5.9, the clearing of the fault at 0.3 seconds does not give the system enough time to regain stability and the generators lose synchronism. The generator at bus 2, which is close to the fault and has a low inertia, has the maximum speed deviation. The bus voltages experience chaotic oscillations and do not re-attain the initial steady state values.

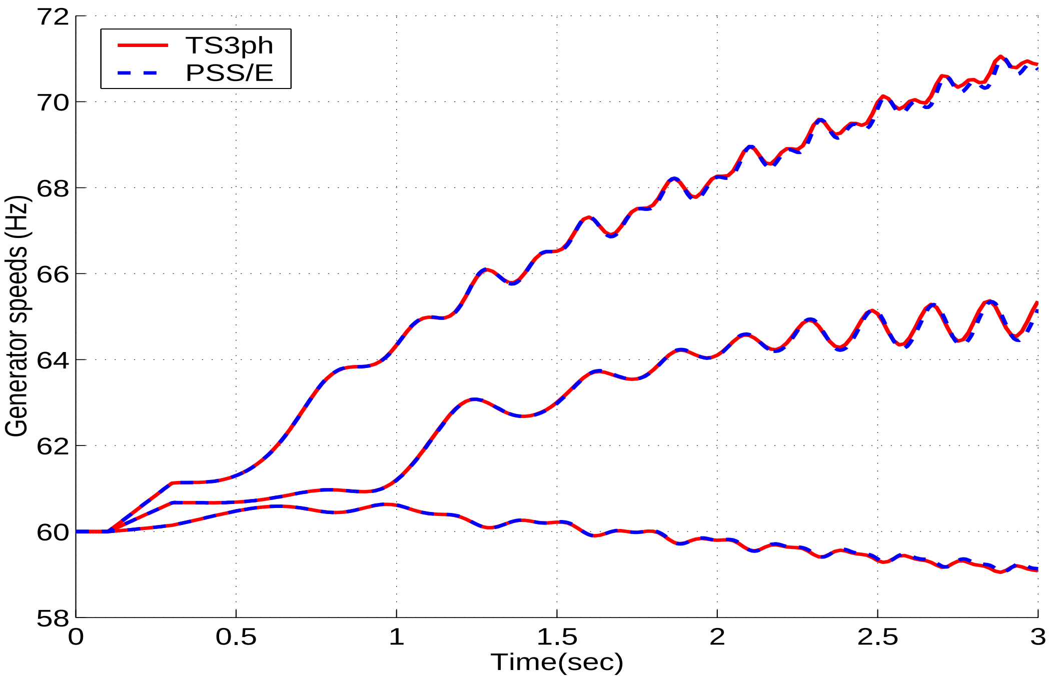


Figure 5.6. Generator speeds for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

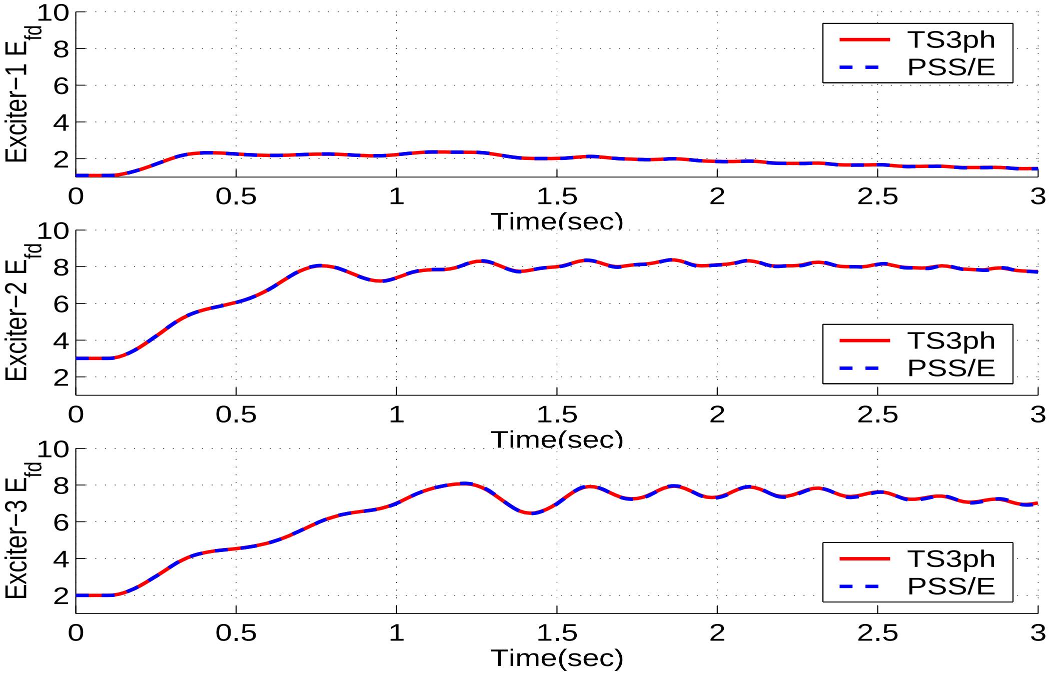


Figure 5.7. Exciter voltages for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

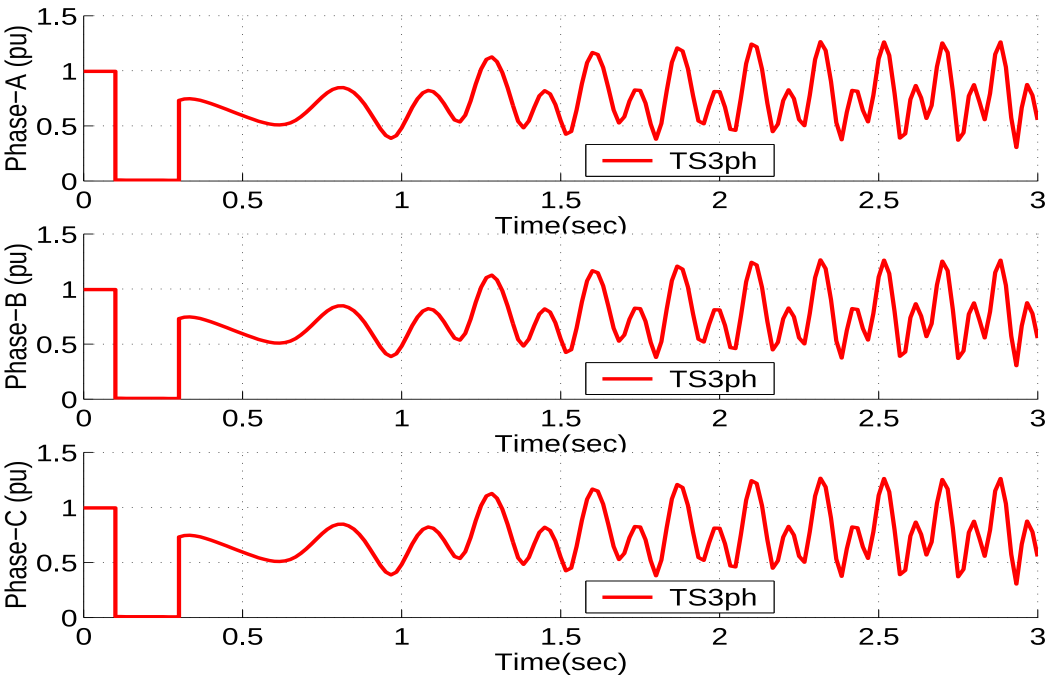


Figure 5.8. Bus 5 three phase phasor voltages for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

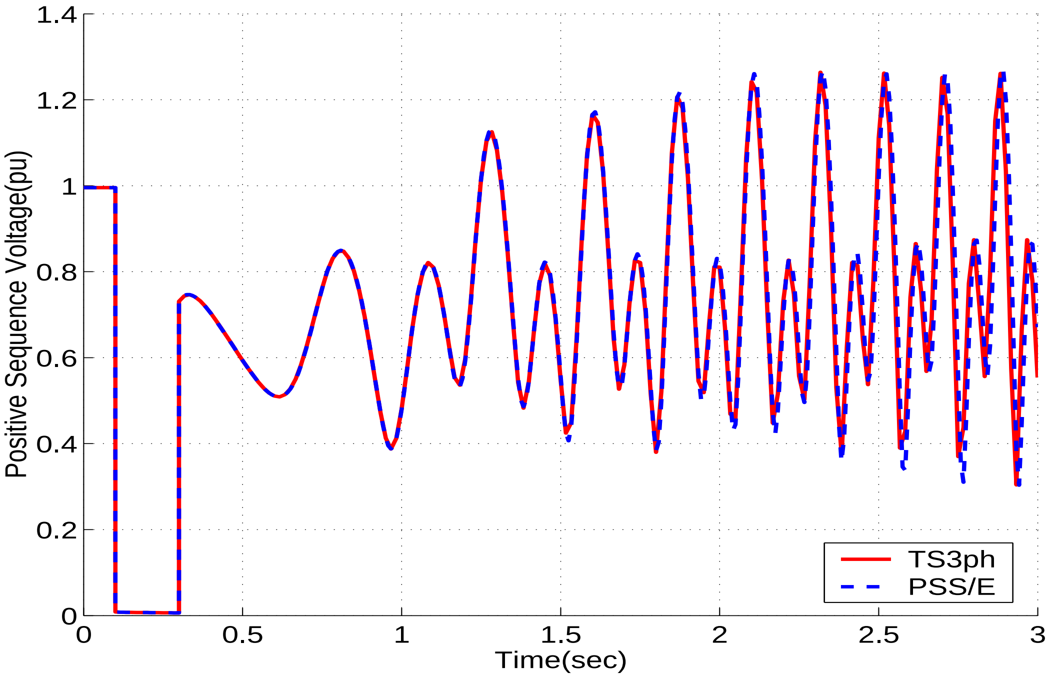


Figure 5.9. Bus 5 positive sequence voltage for a three phase fault on bus 5 from 0.1 sec to 0.3 sec

5.6.2 Single phase faults. Single phase or line-ground faults comprise of about 80% of the total faults on a power system. However, transient stability simulators were mainly developed for assessing the effect of the worst fault scenario, i.e., a three phase fault, on the generator rotor stability. A Three phase fault can be directly simulated in TS assuming it is a balanced fault.

On the other hand, a single phase fault cannot be directly simulated, since TS does not represent all three phases. As the impact of a single phase fault is much less than a three phase fault, a single phase fault is modeled by a larger equivalent positive sequence shunt impedance, than that used for three phase faults. A higher shunt impedance produces a smaller fault current. On the other hand, TS3ph faces no such equivalent modeling since all three phases are represented and faults can be applied on single and multiple phases.

For the single phase test scenario, we used $G_{fault} = 1$ pu in PSS/E and simulated the bus 5 stable case scenario, i.e., the fault is cleared at 0.2 seconds. This larger shunt represents a single-phase fault on bus 5. Next, a single phase fault was applied on phase a at bus 5 and the fault admittance varied so that both PSS/E and TS3ph produced similar positive sequence voltage profiles.

A fault admittance of 0.17 pu was found to match the PSS/E and TS3ph positive sequence voltages as shown in Figure5.11. PSS/E limited to positive sequence models can give information only about the positive sequence voltages, while all the three phase voltages can be simulated in TS3ph as seen in Figure 5.10. The result of the single phase fault is that the voltage on the faulted phase, phase a, drops while the other two phase voltages increase, which cannot be simulated with PSS/E.

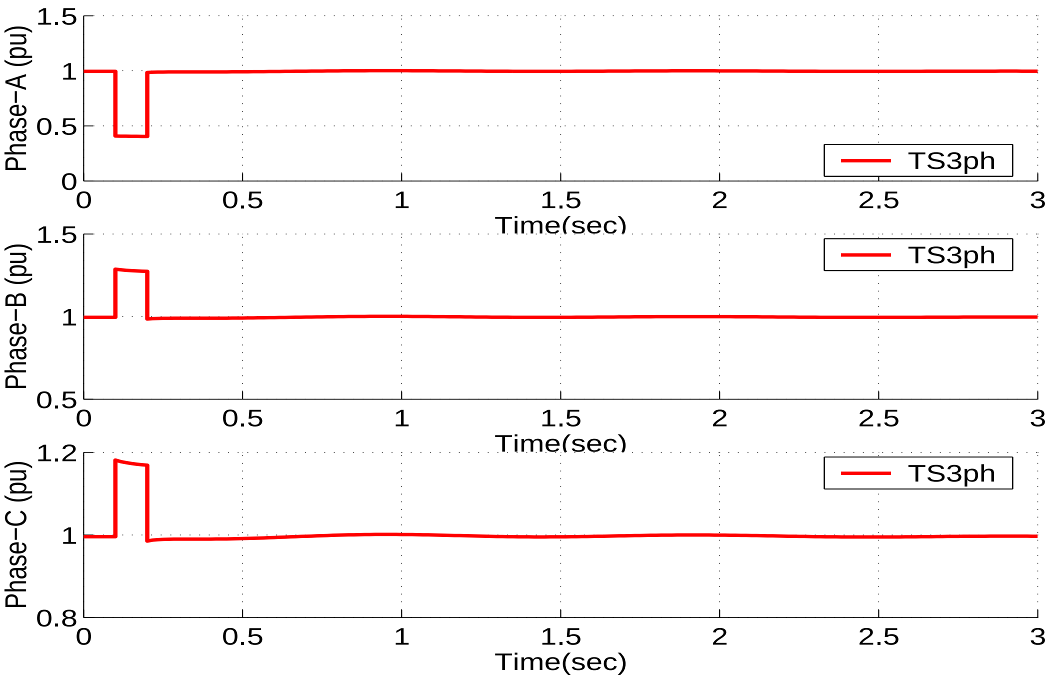


Figure 5.10. Bus 5 three phase voltages for a single phase fault on bus 5 from 0.1 sec to 0.2 sec

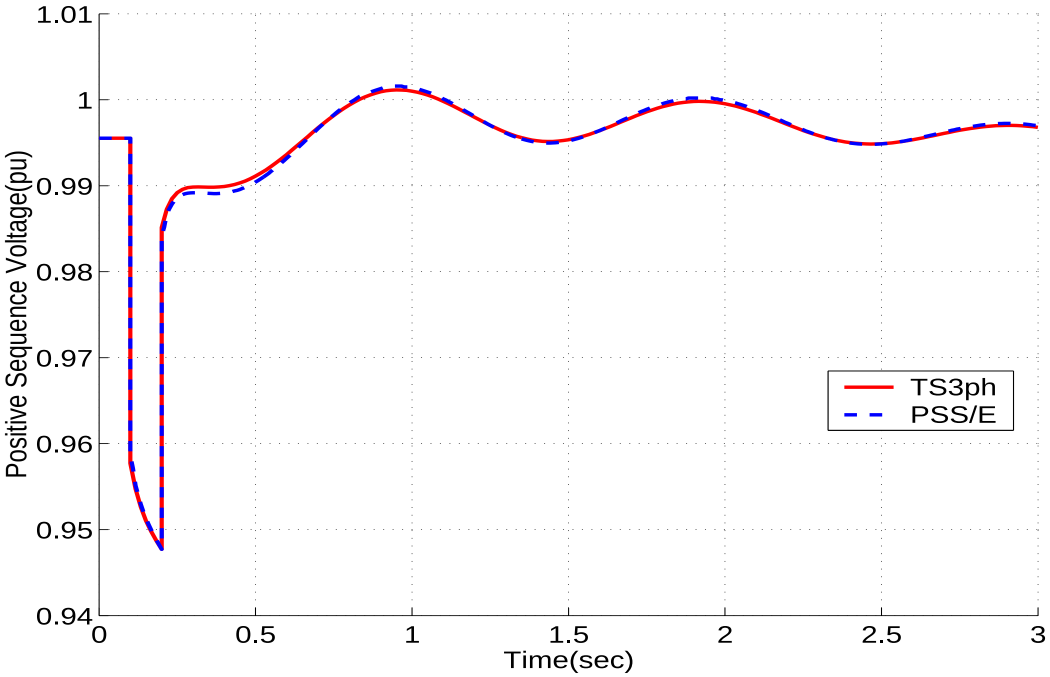


Figure 5.11. Bus 5 positive sequence voltage for a single phase fault on bus 5 from 0.1 sec to 0.2 sec

5.7 Optimizing sequential TS3ph code

From the test runs on different sized systems it was observed that the maximum time spent was on (a) Jacobian computation and (b) linear solves. To speed up the Jacobian computation, we split the jacobian into a linear part and a nonlinear part. The linear part is set before the time stepping begins and only the nonlinear part needs to be updated during the Newton iterations. To optimize the linear solves we experimented with various reordering strategies, direct and iterative linear solvers, and different preconditioners. The results of these experiments are presented in the following subsections.

5.7.1 Jacobian Computation. From the test runs it was observed that setting values in the Jacobian during each Newton iteration is expensive and consumes about 10-15% of the total simulation time for large systems. To speed up the Jacobian computation and also from the observation that the Jacobian for transient stability simulation has more linear terms than nonlinear, we set the linear part of the Jacobian before the time stepping begins and then set only the nonlinear part during the Newton iterations.

Thus, network admittance "Y" matrix, which is the the biggest linear block in the Jacobian, is set before the time stepping begins. Moreover, by rewriting the generator dynamic and static algebraic equations as a sum of linear and nonlinear parts

$$\begin{bmatrix} \dot{x_{gen}} \\ 0 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_{gen} \\ I_{dq} \end{bmatrix} + \begin{bmatrix} R(x_{gen}, I_{dq}, V) \\ R_{Idq}(x_{gen}, V) \end{bmatrix} + \begin{bmatrix} u \\ 0 \end{bmatrix}$$

the linear jacobian part for the generator subsystem equations

$$\begin{bmatrix} I - \frac{\Delta t}{2} A & -\frac{\Delta t}{2} B \\ C & D \end{bmatrix} \tag{5.12}$$

also can be also set before the time stepping begins. Thus, only the nonlinear terms need to be updated during the Newton iterations. Any overlapping elements are also updated in the Jacobian evaluation during the Newton iterations.

Such a strategy resulted in the Jacobian computation time reduced from 10-15% to 2-3 % of the overall CPU time.

5.7.2 Reordering strategies. By reordering the rows and columns of a matrix, it may be possible to reduce the amount of fill-in created by LU factorization, thereby reducing time and storage cost. We experimented with various reordering strategies, available with PETSc, on the test systems to determine the optimal reordering strategy, i.e., the ordering scheme resulting in the least number of nonzeros in the factored matrix. Five reordering schemes were experimented with tested and their abbreviations are given in table 5.2.

Table 5.2. Ordering schemes tested for TS3ph

Abbreviation	Ordering
rcm	Reverse-Cuthill Mckee ordering
qmd	Quotient Minimum degree ordering
natural	Natural Ordering
1wd	One-way dissection
nd	Nested dissection

From the experiments with the reordering strategies on different sized systems, it was found that quotient minimum degree ordering, *qmd*, results in the least number of nonzeros in the factored matrix for the 118 and 1180 bus system, while Reverse Cuthill-Mckee, *rcm*, works out best for the 9 bus system, although *qmd* gives nearly the same performance.

Table 5.3. Results of various reordering schemes for the 9 bus system

Ordering Type	Jacobian Matrix nnz	Factored Matrix nnz	Factor Fill Ratio
natural	1212	1872	1.54455
qmd	1212	1520	1.25413
rcm	**1212**	**1471**	**1.2137**
1wd	1212	1589	1.31106
nd	1212	2175	1.79455

Table 5.4. Results of various reordering schemes for the 118 bus system

Ordering Type	Jacobian Matrix nnz	Factored Matrix nnz	Factor Fill Ratio
natural	21150	90690	4.28794
qmd	**21150**	**29261**	**1.3835**
rcm	21150	38374	1.814374
1wd	21150	36790	1.73948
nd	21150	74576	3.52605

Table 5.5. Results of various reordering schemes for the 1180 bus system

Ordering Type	Jacobian Matrix nnz	Factored Matrix nnz	Factor Fill Ratio
natural	227538	6501246	28.5721
qmd	**227538**	**882397**	**3.87802**
rcm	227538	4332724	19.0418
1wd	227538	3883588	17.0679
nd	227538	4386078	19.2762

5.7.3 Speeding up the linear solver. Newton's method requires the solution of the linear system

$$J(x^i)\Delta x^{i+1} = -F(x^i) \tag{5.13}$$

where i is the iteration count. Solution of 5.13 can be either done by direct or iterative methods.

In direct methods, a gaussian elimination, or factorization, on $J(x^i)$ is done to form lower and upper triangular matrices L and U. Then a forward elimination on L and backward substitution on U is done to find Δx^{i+1}. This 'exact' LU factorization process can be the most time consuming and dominant of all the operations. For our test case systems, the numerical LU factorization process was the most dominant and consumed about 90% of the total time. Hence, we also experimented with incomplete level-based factorization schemes, which differ in the amount of fill-ins added to the factored LU matrices with ILU(0) being the cheapest of all. The results of this experimentation on the different systems are presented in tables 5.6, 5.8, and 5.10.

An alternative to the solution of 5.13 are the iterative Krylov subspace methods with Generalized Minimum Residual algorithm (GMRES) being one of the popular methods. The convergence of the GMRES algorithm depends on the eigenvalues of the operating matrix A and can be slow if the matrix has widely dispersed eigenvalues. Hence, to speed up the GMRES algorithm a preconditioned matrix P^{-1}, where P^{-1} approximates A^{-1}, is generally used. PETSc provides a variety of preconditioners and we experimented with level-based incomplete factorization, ILU(k), as a preconditioner to optimize the iterative GMRES linear solution process.

From our experiments, we found that the most time consuming part is the numerical factorization phase. Hence, we used a strategy of lagging the factorization (for direct method) and preconditioners (for the iterative method) to save time spent on factorization. We experimented with a strategy of lagging the numerical factorization, called '*lag*' in the tables, by doing it only when there is a change in the Jacobian matrix structure such as that caused when the disturbance is inserted or removed. For all the other time steps, the previous factored matrix is reused.

All these experiments were run with optimization level -O3 on an 2.53 GHz, 4 core, 12 GB RAM, 8 MB cache Intel Xeon Processor. The total simulation time was 3 seconds with a time step of 0.016667 seconds, i.e., 1 cycle. In all these simulations a three phase fault was applied at 0.1 seconds and removed at 0.2 seconds. In tables 5.6 - 5.11 the reordering strategy used is *qmd* unless any other strategy is mentioned and xxx denotes that the linear solver did not converge when the fault was placed.

Table 5.6. TS3ph timing results for 118 bus system with direct linear solution schemes

Factorization Type and ordering	Total time (sec)	Timeloop (sec)
LU	8.78E-01	4.57E-01
LU with nd	2.16E+00	1.79E+00
LU with lag	4.08E-01	3.65E-01
LU with lag and nd	5.12E-01	4.28E-01
ILU(0)	xxx	xxx
ILU(0) with nd	6.39E+00	6.11E+00
ILU with factor level 2	6.09E+00	5.80E+00
ILU with factor level 2 with nd	6.29E+00	6.01E+00
ILU with factor level 2, with lag	2.07E+00	1.83E+00
ILU with factor level 2, with lag, and with nd	2.26E+00	2.02E+00
ILU with factor level 4	8.71E-01	6.68E-01
ILU with factor level 4 with nd	6.30E+00	6.05E+00
ILU with factor level 4, with lag	4.07E-01	3.64E-01
ILU with factor level 4, with lag, and with nd	2.26E+00	2.02E+00
ILU with factor level 6	4.92E-01	4.48E-01
ILU with factor level 6, with nd	6.35E+00	6.10E+00
ILU with factor level 6, with lag	**4.02E-01**	**3.60E-01**
ILU with factor level 6, with lag, and with nd	2.30E+00	2.01E+00

Table 5.7. TS3ph timing results for 118 bus system with GMRES

Preconditioner type and ordering	Total time (sec)	Timeloop (sec)
LU	5.97E-01	4.86E-01
LU with nd	2.14E+00	1.94E+00
LU with lag	3.82E-01	3.40E-01
LU with lag and nd	5.05E-01	4.61E-01
ILU(0)	1.62E+00	1.37E+00
ILU(0) with nd	1.29E+00	9.68E-01
ILU with factor level 2	1.13E+00	8.40E-01
ILU with factor level 2 with nd	1.26E+00	9.74E-01
ILU with factor level 2, with lag	6.74E-01	5.91E-01
ILU with factor level 2, with lag, and with nd	9.78E-01	7.35E-01
ILU with factor level 4	6.10E-01	5.66E-01
ILU with factor level 4 with nd	1.30E+00	9.74E-01
ILU with factor level 4, with lag	3.88E-01	3.46E-01
ILU with factor level 4, with lag, and with nd	8.97E-01	7.34E-01
ILU with factor level 6	5.61E-01	5.17E-01
ILU with factor level 6, with nd	1.29E+00	9.68E-01
ILU with factor level 6, with lag	**3.87E-01**	**3.45E-01**
ILU with factor level 6, with lag, and with nd	8.30E-01	7.39E-01

Table 5.8. TS3ph timing results for 1180 bus system with direct linear solver schemes

Factorization Type and ordering	Total time (sec)	Timeloop (sec)
LU	6.83E+01	6.76E+01
LU with nd	1.32E+03	1.32E+03
LU with lag	1.29E+01	1.24E+01
LU with lag and nd	3.51E+01	3.46E+01
ILU(0)	xxx	xxx
ILU(0) with nd	7.97E+01	7.89E+01
ILU with factor level 2	xxx	xxx
ILU with factor level 2 with nd	7.22E+01	7.16E+01
ILU with factor level 2, with lag	xxx	xxx
ILU with factor level 2, with lag, and with nd	2.42E+01	2.36E+01
ILU with factor level 4	xxx	xxx
ILU with factor level 4 with nd	7.24E+01	7.19E+01
ILU with factor level 4, with lag	xxx	xxx
ILU with factor level 4, with lag, and with nd	2.41E+01	2.36E+01
ILU with factor level 6	1.20E+02	1.19E+02
ILU with factor level 6, with nd	7.70E+01	7.64E+01
ILU with factor level 6, with lag	**1.20E+01**	**1.15E+01**
ILU with factor level 6, with lag, and with nd	2.42E+01	2.36E+01

Table 5.9. TS3ph timing results for 1180 bus system with GMRES

Preconditioner type and ordering	Total time (sec)	Timeloop (sec)
LU	6.99E+01	6.92E+01
LU with nd	1.26E+03	1.26E+03
LU with lag	9.45E+00	8.93E+00
LU with lag and nd	3.29E+01	3.22E+01
ILU(0)	1.68E+01	1.62E+01
ILU(0) with nd	1.43E+01	1.37E+01
ILU with factor level 2	1.27E+01	1.20E+01
ILU with factor level 2 with nd	1.41E+01	1.35E+01
ILU with factor level 2, with lag	8.54E+00	7.94E+00
ILU with factor level 2, with lag, and with nd	1.10E+01	1.04E+01
ILU with factor level 4	1.95E+01	1.89E+01
ILU with factor level 4 with nd	1.40E+01	1.34E+01
ILU with factor level 4, with lag	8.85E+00	8.26E+00
ILU with factor level 4, with lag, and with nd	1.09E+01	1.04E+01
ILU with factor level 6	3.46E+01	3.39E+01
ILU with factor level 6, with nd	1.42E+01	1.36E+01
ILU with factor level 6, with lag	**7.78E+00**	**7.23E+00**
ILU with factor level 6, with lag, and with nd	1.10E+01	1.04E+01

Table 5.10. TS3ph timing results for 2360 bus system with direct linear solver schemes

Factorization Type and ordering	Total time (sec)	Timeloop (sec)
LU	3.14E+03	3.13E+03
LU with nd	xxx	xxx
LU with lag	6.60E+01	6.48E+01
LU with lag and nd	2.40E+02	2.39E+02
ILU(0)	xxx	xxx
ILU(0) with nd	xxx	xxx
ILU with factor level 2	xxx	xxx
ILU with factor level 2 with nd	xxx	xxx
ILU with factor level 2, with lag	xxx	xxx
ILU with factor level 2, with lag, and with nd	xxx	xxx
ILU with factor level 4	xxx	xxx
ILU with factor level 4 with nd	xxx	xxx
ILU with factor level 4, with lag	xxx	xxx
ILU with factor level 4, with lag, and with nd	xxx	xxx
ILU with factor level 6	xxx	xxx
ILU with factor level 6, with nd	xxx	xxx
ILU with factor level 6, with lag	**4.71E+01**	**4.60E+01**
ILU with factor level 6, with lag, and with nd	xxx	xxx

Table 5.11. TS3ph timing results for 2360 bus system with GMRES

Preconditioner type and ordering	Total time (sec)	Timeloop (sec)
LU	1.94E+03	1.94E+03
LU with nd	1.93E+04	1.93E+04
LU with lag	6.70E+01	6.59E+01
LU with lag and nd	2.90E+02	2.89E+02
ILU(0)	4.57E+01	4.47E+01
ILU(0) with nd	4.48E+01	4.37E+01
ILU with factor level 2	4.95E+01	4.84E+01
ILU with factor level 2 with nd	4.47E+01	4.37E+01
ILU with factor level 2, with lag	3.63E+01	3.51E+01
ILU with factor level 2, with lag, and with nd	3.55E+01	3.44E+01
ILU with factor level 4	1.83E+02	1.81E+02
ILU with factor level 4 with nd	4.47E+01	4.37E+01
ILU with factor level 4, with lag	4.01E+01	3.90E+01
ILU with factor level 4, with lag, and with nd	3.56E+01	3.44E+01
ILU with factor level 6	6.89E+02	6.88E+02
ILU with factor level 6, with nd	4.48E+01	4.37E+01
ILU with factor level 6, with lag	4.43E+01	4.33E+01
ILU with factor level 6, with lag, and with nd	**3.54E+01**	**3.44E+01**

5.7.4 Summary of TS3ph code optimization results. For the given three test systems, the following can be summarized

1. *qmd* reordering strategy was the best for the three test systems giving the least number of nonzeros in the factored matrix.

2. An 'exact' LU factorization or preconditioner is the most expensive amongst all the tested linear solver strategies.

3. Doing a numerical factorization only during the fault on/off time steps, i.e., *lagging* the factorization until a structural change to the Jacobian, resulted in considerable speed up for both the direct method and iterative GMRES solver.

4. With direct methods the incomplete LU factorization methods may fail to converge. For the 2360 bus system, all the tested incomplete LU factorization direct method schemes failed to converge.

5. Incomplete LU as a preconditioner for GMRES does not face any convergence problems.

6. For all the three test cases the fastest run times were obtained with GMRES.

7. Incomplete LU with higher factorization levels were not tested but may result in faster run times for smaller systems.

8. For all the test systems, GMRES with ILU(6) preconditioner and lagging strategy was the fastest.

CHAPTER 6

PROPOSED IMPLICITLY COUPLED TSEMT SIMULATOR

This chapter discusses the theoretical basis for the implicitly coupled electromechanical and electromagnetic transients simulator that has been developed in this research work. The motivation, differences between the existing hybrid simulators and proposed implicitly coupled TSEMT simulator, network equivalents, and the implicitly coupled solution approach are discussed. Following this, the strategy we adopt for a combined electromechanical and electromagnetic transients simulation is explained.

6.1 Motivation

The development of our proposed implicitly coupled TSEMT simulator is based on the following three observations of the existing hybrid simulators.

6.1.1 Constant or extrapolated external system equivalent. In the existing interaction protocols TS and EMT exchange the data at regular intervals. At the beginning of each TS time step, TS transfers the external system equivalent to EMT, and EMT progresses to the next TS time step where it updates the detailed system equivalent and passed it back to TS. It is to be noted here that the external system equivalent is not updated when EMT is running, i.e., it is held constant for all the EMT time steps within a TS time step. This equivalent can be also derived from some extrapolated history data, but it either way, it may not accurately predict the conditions at the next TS time step.

Moreover, it is to be noted that the evolution of the EMT solutions is done using the constant external equivalent and, at the end of each TS time step EMT computes the detailed system equivalent. This detailed system equivalent passed back to TS is calculated from the combined EMT solutions that used a constant external system equivalent. While such an approach would be sufficient if the TS system is evolving slowly, i.e., there is a small difference between the voltages and currents at two consecutive time steps, for large changes this approach may not be suitable.

We present simulation results for the serial interaction protocol on the test WECC 9-bus system to justify our argument. The detailed system consists of buses 7, 8, and 9 with two transmission lines 7-8 and 8-9 and a load modeled as constant impedance on bus 8. Buses 7 and 9 form the boundary buses.

As a disturbance scenario, a three phase fault is placed on bus 8 in the interior of the detailed system at 0.1 seconds and removed at 0.2 seconds. All the generators are in the external system and modeled as constant voltage sources. The external system equivalent for the EMT simulator is a fundamental frequency Thevenin equivalent.

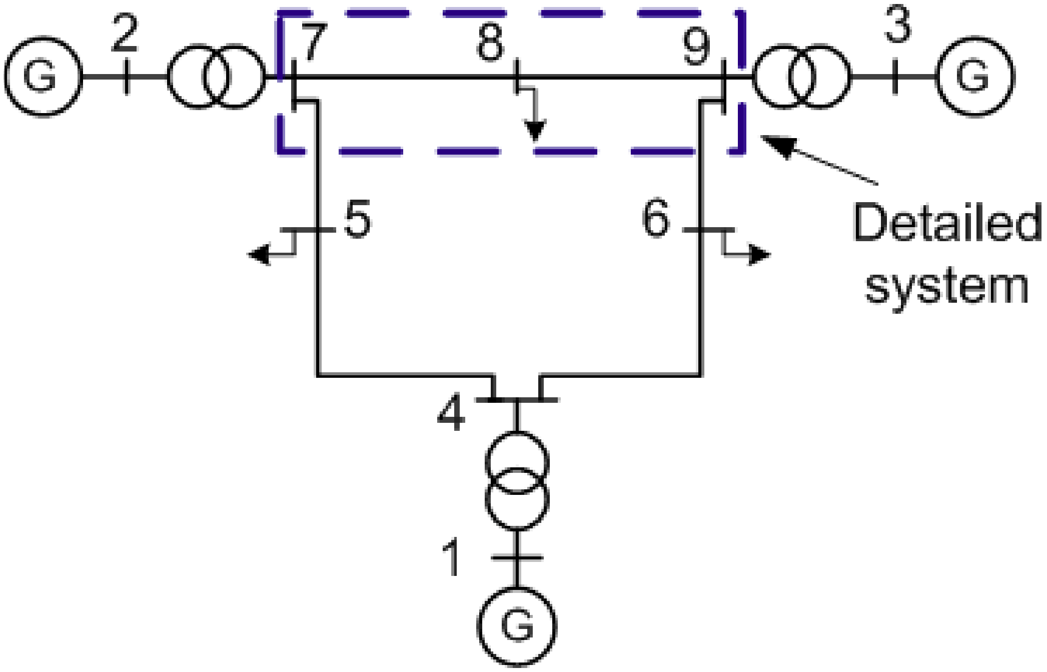


Figure 6.1. 9-bus system with buses 7,8,9 modeled in EMT

As seen in figure 6.2, for this test case, the hybrid simulator results are comparable to EMT. Note here that the boundary buses, 7 and 9, are connected to the generator buses, 2 and 3. These generator bus voltages do not change before, during, or after the fault since the generators are modeled as constant voltage sources. The presence of the these constant generator bus voltages result in the Thevenin equivalent voltages to be more or less constant, i.e., the external system equivalent is nearly constant and this is true even after the fault is applied. The external system for this test scenario can be regarded as 'strong' since the effect of the fault does not cause considerable change in the external system voltages resulting in a nearly constant external system for EMT.

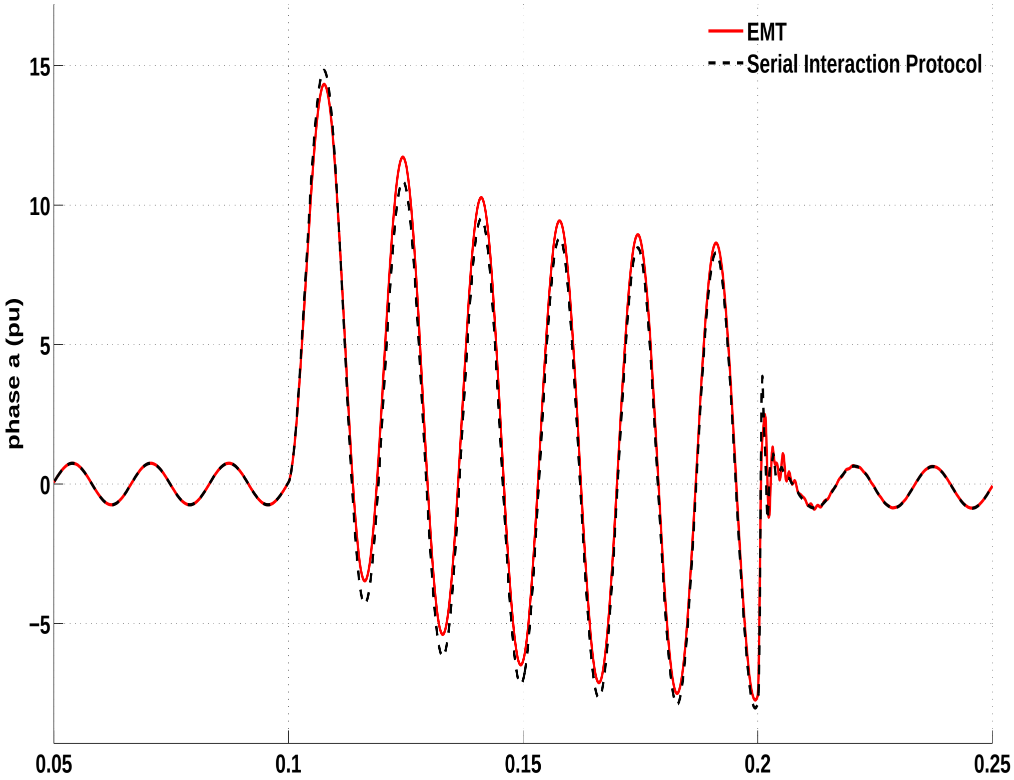


Figure 6.2. Bus 7 phase a boundary currents with generators modeled as constant voltage sources

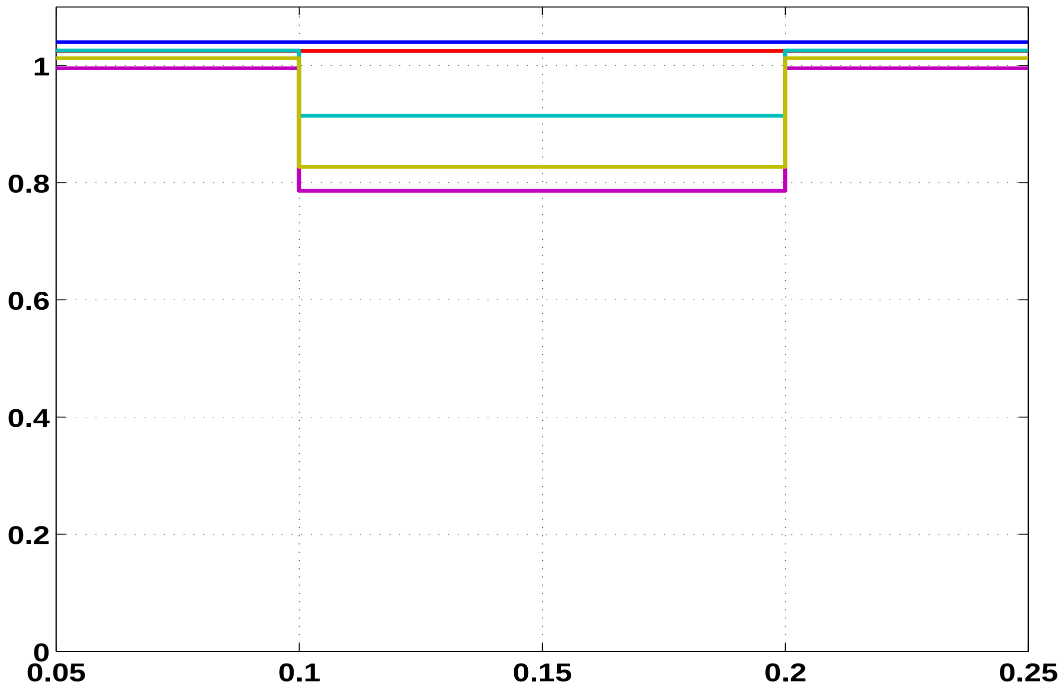


Figure 6.3. External system positive sequence bus voltages with generators modeled as constant voltage sources

Next, we replaced the constant voltage source model by GENROU for all three generators. A generator bus voltage may no longer be constant now since it depends on the interaction of the network and the generator dynamics. The results from this test are shown in figure 6.4.

As seen in 6.4, the serial interaction protocol produces incorrect results and fails to converge. The Thevenin equivalent for this scenario is not constant when the fault is applied since the generator bus voltages are not constant voltage sources as in the previous test. In this case, the generator bus voltages drop when the fault is applied.

However, EMT uses the ‘constant pre-fault’ Thevenin equivalent voltage for the time steps immediately after the fault is applied. Due to the incorrect Thevenin equivalent voltage, errors are introduced in the EMT solutions which get propagated to TS at the next data exchange. The accumulation of these errors results in the non-convergent behavior at a future time step. The external system in this case can be considered as 'weak' since the fault on bus 8 causes considerable change in the external system voltages resulting in a large change in the Thevenin equivalent voltage.

Figure 6.5 shows the zoomed-in plot of the serial interaction protocol for time steps immediately following the fault. The boundary current in figure 6.5 for the immediate cycle after the fault is the same as that obtained in the previous test case, which is incorrect. As such, errors are introduced in the EMT solutions which get passed to TS at the next interchange and eventually lead to the non-convergence behavior.

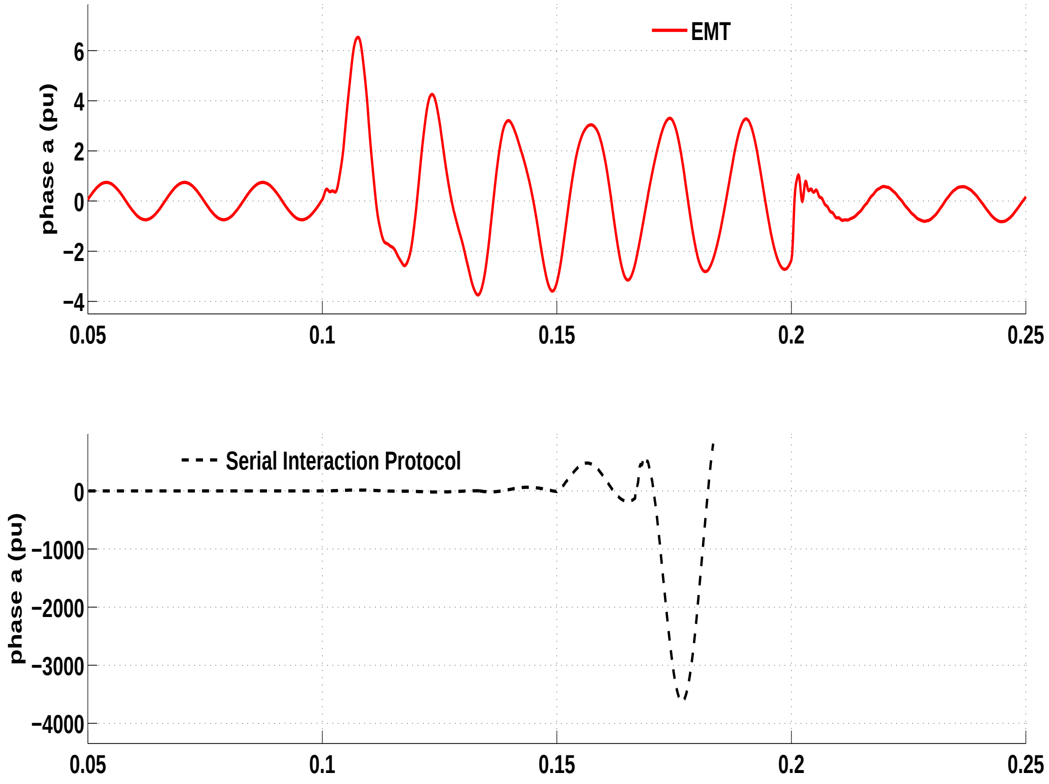


Figure 6.4. Non-convergent behavior of the serial interaction protocol

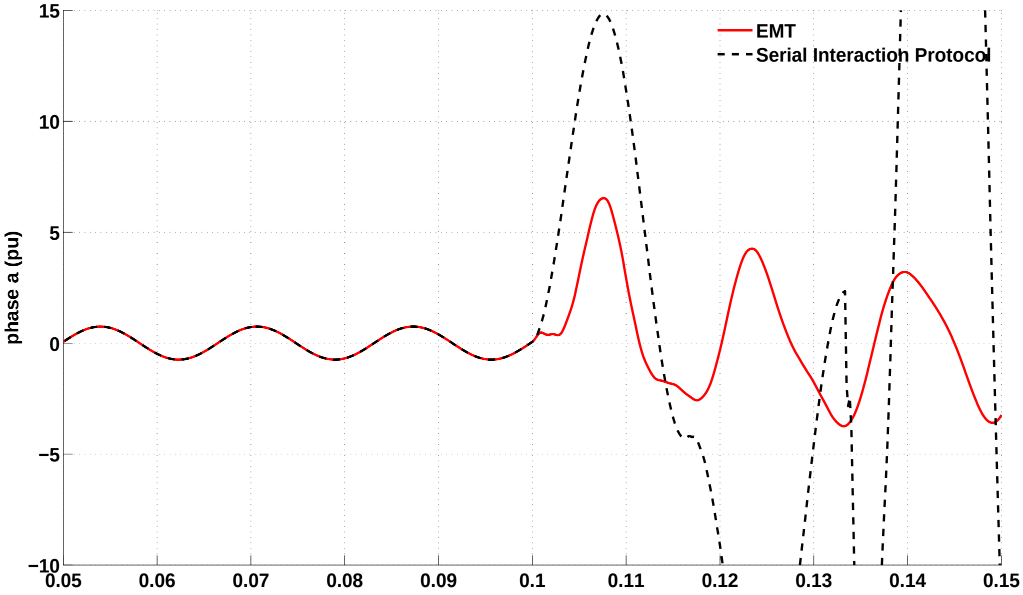


Figure 6.5. Zoomed-in plot of the serial interaction protocol for non-convergent behavior

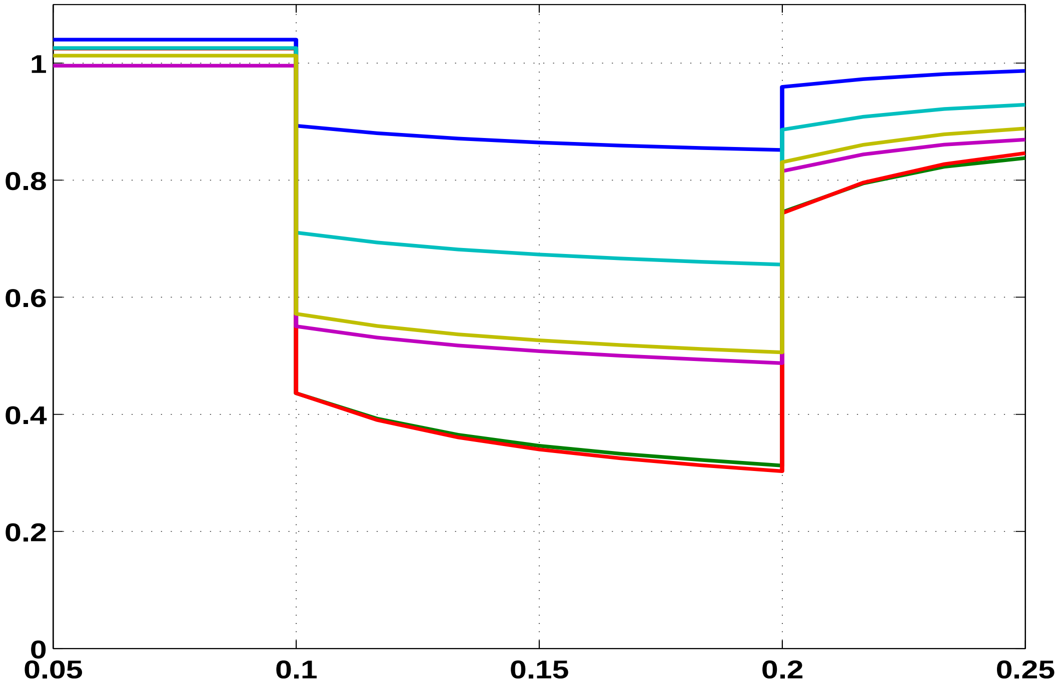
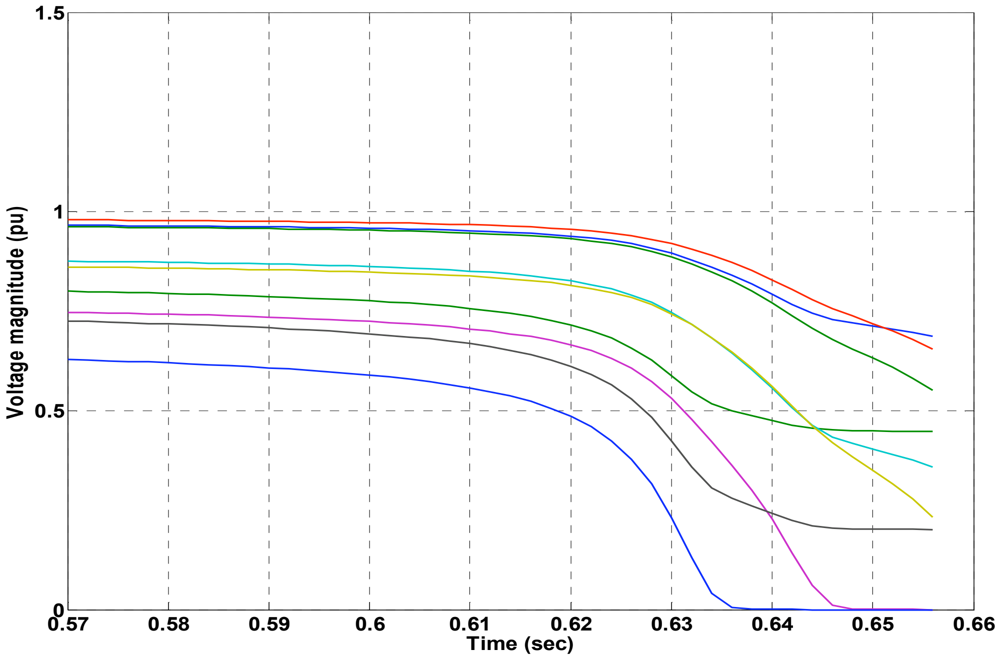


Figure 6.6. External system positive sequence bus voltages with generators modeled with GENROU model

Although the parallel interaction protocol wasn't implemented, it can be shown that it also suffers from the same drawback as the serial protocol,i.e., if there is a large change in the external system then the hybrid simulator will produce inaccurate results and may fail to converge.

Furthermore, the serial or the parallel interaction protocol may not work for a voltage collapse scenario. Figure 6.7 shows the TS voltage plots for a voltage collapse scenario. The voltage collapse plots have a linear portion in which the voltages are decreasing gradually until 0.62 seconds and then there is a sudden drop in the voltages. This is a characteristic of voltage collapse scenarios where the voltages tend to decrease gradually and then collapse almost instantaneously at some point,e.g., in Figure 6.7 after 0.62 seconds. If the above voltage collapse scenario were to be reproduced by a hybrid simulator using a serial or a parallel interaction protocol then having a constant or extrapolated external system equivalent might be sufficient



!

Figure 6.7. Voltage collapse plots

when the voltages are decreasing gradually. However when the voltages have a large difference between consecutive time steps this approach will suffer.

6.1.2 Explicit TS and EMT coupling and no iterations.

The existing interaction protocols were developed to 'interface' existing TS and EMT programs. Hence, there exists an explicit coupling between TS and EMT where the data is exchanged after their individual solutions are computed. Each hybrid simulator time step ends with EMT transferring the detailed system equivalent to TS and then the hybrid simulator proceeds to the next TS time step. Note that no iterations are done between TS and EMT to check if the solutions at each TS and EMT boundary are consistent. Having no iterations is probably sufficient when the external system equivalent does not change much as seen from the results of 6.2, and it may be adequate for the gradually changing voltage profile in Figure 6.7. However, near voltage collapse, iterations would be needed to update the external system equivalent repeatedly. Due to the explicit coupling, more iterations would be required and the solution still might diverge.

6.1.3 Balanced external system equivalent.

One of the major assumptions in the TS is that the transmission network is always balanced. Hence a positive sequence network suffices for the analysis. For the hybrid simulators, such an assumption results in using only a balanced external system equivalent for EMT.

A balanced external system equivalent might be sufficient if the EMT system is balanced prior, during, or after the transient but when the EMT system is unbalanced, the assumption that the external system equivalent is still balanced is clearly invalid. An example scenario is a single-phase fault at the boundary bus that causes an unbalance in the boundary bus currents and voltages. For this example scenario, a balanced system equivalent is unrealistic.

6.2 Differences between proposed TSEMT and existing hybrid simulators

The implicitly coupled TSEMT simulator developed in this research work is based on the following two major differences

1. **Implicitly coupled solution approach:** Instead of coupling TS and EMT at the application level, we propose to couple them at the equation solution level. To combine the two sets of equations with their different time steps, and ensure that the TS and EMT solutions are consistent, the equations for TS and coupled-in-time EMT equations are solved simultaneously in a single large system of equations. While computing a single time step of the TS equations, a simultaneous calculation of several time steps of the EMT equations is undertaken. For the remainder of this thesis, the implicitly coupled TS and EMT simulator will be refered to as TSEMT.

2. **Three phase external system:** This research work also proposes a full three-phase phasor model of the external system. The proposed three-phase TS simulator, TS3ph, is used for the external system in the implicitly coupled TSEMT simulator.

6.3 TS3ph and EMT coupling

Network equivalents and waveform conversion form the coupling between TS3ph and EMT. For the proposed TSEMT simulator, we use a Thevenin equivalent of the external system and fundamental frequency phasor current source injection as the coupling between TS3ph and EMT. The Thevenin equivalent connects the EMT to TS3ph and the fundamental frequency phasor current source injection connects TS3ph to EMT as shown in 6.8.

The subscripts used for three phase formulation, $_{abc}$, are omitted in the follow-

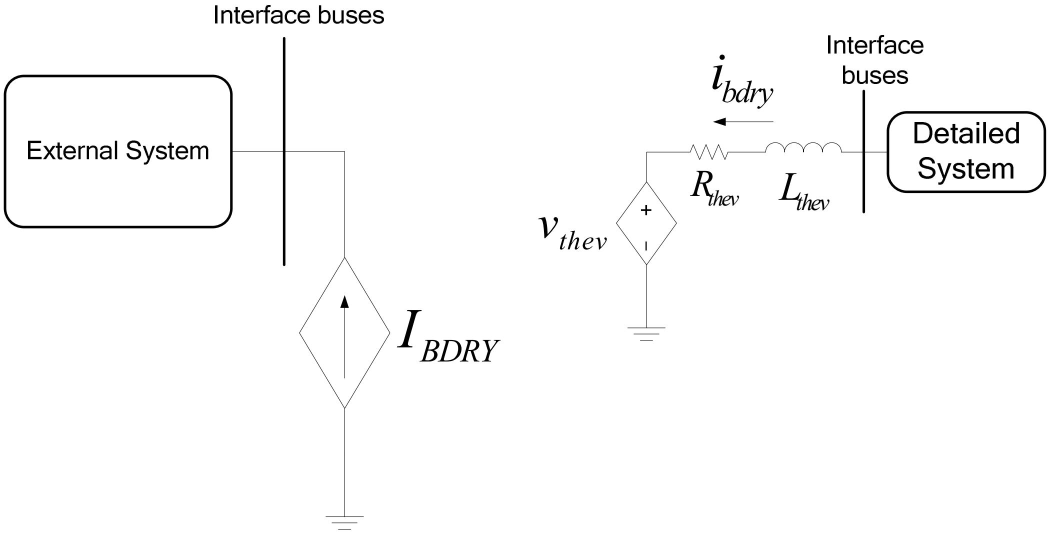


Figure 6.8. Equivalent networks for detailed and external system

ing subsections for clarity, although the three phase formulation is still used in the implicitly coupled TSEMT simulator.

6.3.1 External system equivalent. We use a fundamental frequency Thevenin equivalent of the external system network as the external system equivalent for EMT. This Thevenin equivalent can be derived from the Z_{bus} matrix of the external system network. The voltage current relationship using the Z_{bus} matrix is

$$\begin{bmatrix} Z_{int} & Z_{int,bdry} \\ Z_{bdry,int} & Z_{bdry,bdry} \end{bmatrix} \begin{bmatrix} I_{int} \\ I_{bdry} \end{bmatrix} = \begin{bmatrix} V_{int} \\ V_{bdry} \end{bmatrix} \tag{6.1}$$

Here the subscripts *int* and *bdry* indicate the interior and boundary buses respectively. Doing some algebraic manipulation, the boundary bus voltages can be written as

$$V_{bdry} = \overbrace{Z_{bdry,int} Z_{int,int}^{-1} V_{int}}^{\text{Thevenin voltage source}} - \overbrace{\left(Z_{bdry,int} Z_{int,int}^{-1} Z_{int,bdry} - Z_{bdry,bdry}\right)}^{\text{Thevenin equivalent impedance}} I_{bdry} \tag{6.2}$$

Thus for the detailed system, the external system is represented as a Thevenin voltage behind a thevenin impedance. The Thevenin impedance is kept constant throughout the simulation, unless there is a topological change in the external system. Only the Thevenin voltage V_{thev} needs to be updated at each time step. For EMT, the Thevenin impedance Z_{thev} is converted to a series branch with resistance R_{thev} and inductance L_{thev} compted at the fundamental frequency. Thus EMT is coupled to TS3ph via a series RL branch and a Thevenin equivalent voltage source as shown in 6.8. Since EMT uses instantaneous voltages, phasor voltage V_{thev} needs to be converted to instantaneous waveform v_{thev}. This conversion is done by a fundamental frequency sine wave generator.

The inclusion of the external system equivalent in EMT introduces an additional set of differential equations

$$L_{thev}\frac{di_{bdry}}{dt} = v_{thev} - R_{thev}i_{bdry} - v_{bdry} \tag{6.3}$$

where v_{thev} is the instantaneous Thevenin voltage, i_{bdry} is the current flowing out through the detailed system boundary buses, and v_{bdry} is the instantaneous boundary bus voltage.

The current injection at the boundary buses also modifies the differential equations for the EMT network voltages and the modified equations are as follows

$$C\frac{dv}{dt} = A_{line}i_{ser} + A_{gen}i_{gen} - A_{load}i_{load} + A_{bdry}i_{bdry} \tag{6.4}$$

Here, A_{bdry} is the incidence matrix mapping the boundary currents onto the EMT network buses.

6.3.2 Detailed system equivalent. The detailed system equivalent used by the TS portion of our proposed TSEMT simulator is a fundamental frequency phasor current injection at the boundary buses. This phasor current injection is computed via a Fourier analysis of EMT boundary bus currents i_{bdry} over a running window of one cycle of fundamental frequency.

As found in the existing literature, there are two methods for converting instantaneous quantities to phasor: curve fitting and Fourier analysis. While curve fitting approach uses a least square solution method for computing the phasor, Fourier analysis uses an averaging or integration scheme. Our rationale for choosing a Fourier analysis is that it can be easily incorporated in the numerical integration scheme and does not introduce any additional variables as required for a curve fitting process. The phasor current injections at the external system boundary buses can be written in rectangular form by Fourier analysis as

$$I_{BDRY,D}(t + \Delta t_{TS}) = \frac{2}{T} \int_{\tau=t}^{t+\Delta t_{TS}} i_{bdry}(\tau) \sin(\omega \tau) d\tau \tag{6.5}$$

$$I_{BDRY,Q}(t + \Delta t_{TS}) = \frac{2}{T} \int_{\tau=t}^{t+\Delta t_{TS}} i_{bdry}(\tau) \cos(\omega \tau) d\tau \tag{6.6}$$

There are no additional equations needed for the detailed system equivalent I_{BDRY}. However the TS3ph network equations needed to be modified to include this current injection.

$$YV = I_{gen} - I_{load} + I_{BDRY} \tag{6.7}$$

An incidence matrix can be used to map the boundary current injections onto the TS network buses or I_{BDRY} can be a vector with nonzero values only at the boundary buses. The implicitly coupled TSEMT simulator uses the later approach.

6.4 Implicitly coupled solution approach

Before delving into the details of the proposed implicitly coupled solution approach for power system dynamics, let us define a few terms and equations:

- $X_{TS} \equiv$ vector of dynamic variables for TS
- $V_{TS} \equiv$ vector of network voltages for TS
- $x_{emt} \equiv$ vector of EMT dynamic variables

- $i_{bdry} \equiv$ vector of EMT boundary currents

The algebraic variables for EMT and the generator stator currents I_{dq} are skipped for ease of explanation, but the reader can refer to Chapter 3 and 4 for details.

In compact form, the TS3ph system DAE model equations are

$$\begin{aligned} \frac{dX_{TS}}{dt} &= F(X_{TS}, V_{TS}) \\ 0 &= G(X_{TS}, V_{TS}) \end{aligned} \tag{6.8}$$

and the equations for EMT are

$$\frac{dx_{EMT}}{dt} = f(x_{EMT}) \tag{6.9}$$

Adding the coupling, the equations for TS3ph and EMT in compact form are

$$\begin{aligned} \frac{dX_{TS}}{dt} &= F(X_{TS}, V_{TS}) \\ 0 &= G(X_{TS}, V_{TS}, I_{BDRY}) \\ \frac{dx_{EMT}}{dt} &= f_1(x_{EMT}, i_{bdry}) \\ \frac{di_{bdry}}{dt} &= f_2(x_{EMT}, i_{bdry}, v_{thev}) \end{aligned} \tag{6.10}$$

Discretizing the TS equations with the TS time step and EMT equations with EMT time step and using an implicit trapezoidal integration scheme, the complete set of equations to solve at each TS time step are

$$\text{TS3ph} \begin{cases} X_{TS}(t_{N+1}) - X_{TS}(t_N) - \dfrac{\Delta t_{TS}}{2}(F(X_{TS}(t_{N+1}), V_{TS}(t_{N+1}))+ \\ \qquad F(X_{TS}(t_N), V_{TS}(t_N))) = 0 \\ G(X_{TS}(t_{N+1}), V_{TS}(t_{N+1}), I_{BDRY}(t_{N+1})) = 0 \end{cases} \tag{6.11}$$

$$\text{EMT}\left\{\begin{aligned}
& x_{EMT}(t_{n+1}) - \frac{\Delta t_{EMT}}{2} f_1(x_{EMT}(t_{n+1}), i_{bdry}(t_{n+1})) \\
& \; - \; x_{EMT}(t_n) - \frac{\Delta t_{EMT}}{2} f_1(x_{EMT}(t_n), i_{bdry}(t_n)) = 0 \\
& i_{bdry}(t_{n+1}) - \frac{\Delta t_{EMT}}{2} f_2(x_{EMT}(t_{n+1}), i_{bdry}(t_{n+1}), v_{thev}(t_{n+1})) \\
& \; - \; i_{bdry}(t_n) - \frac{\Delta t_{EMT}}{2} f_2(x_{EMT}(t_n), i_{bdry}(t_n), v_{thev}(t_n)) = 0 \\
& x_{EMT}(t_{n+2}) - \frac{\Delta t_{EMT}}{2} f_1(x_{EMT}(t_{n+2}), i_{bdry}(t_{n+2})) \\
& \; - x_{EMT}(t_{n+1}) - \frac{\Delta t_{EMT}}{2} f_1(x_{EMT}(t_{n+1}), i_{bdry}(t_{n+1})) = 0 \\
& i_{bdry}(t_{n+2}) - \frac{\Delta t_{EMT}}{2} f_2(x_{EMT}(t_{n+2}), i_{bdry}(t_{n+2}), v_{thev}(t_{n+2})) \\
& \; - \; i_{bdry}(t_{n+1}) - \frac{\Delta t_{EMT}}{2} f_2(x_{EMT}(t_{n+1}), i_{bdry}(t_{n+1}), v_{thev}(t_{n+1})) \\
= 0 & \\
& \vdots \\
& \vdots \\
& i_{bdry}(t_{n+k}) - \frac{\Delta t_{EMT}}{2} f_2(x_{EMT}(t_{n+k}), i_{bdry}(t_{n+k}), v_{thev}(t_{n+k})) \\
& \; - \; i_{bdry}(t_{n+k-1}) - \frac{\Delta t_{EMT}}{2} f_2(x_{EMT}(t_{n+k-1}), \\
& i_{bdry}(t_{n+k-1}), v_{thev}(t_{n+k-1})) = 0
\end{aligned}\right. \tag{6.12}$$

where

$$\begin{aligned}
& I_{BDRY}(t_{N+1}) = h_{EMT->TS3ph}(i_{bdry}(t_{n+1}), i_{bdry}(t_{n+2}), \ldots, i_{bdry}(t_{n+k})) \\
& (v_{thev}(t_{n+1}), v_{thev}(t_{n+2}), \ldots, v_{thev}(t_{n+k})) \\
& \qquad = h_{TS3ph->EMT}(V_{thev,TS}(t_N), V_{thev,TS}(t_{N+1}))
\end{aligned}$$

represents the coupling between TS3ph and EMT.

6.5 Dimensions of the TSEMT problem

We provide some representative numbers here to get an idea about the overall size of the TSEMT problem to be solved. Suppose the external system consists of

1000 buses, 2000 transmission lines, 1000 loads, and 300 generators modeled with GENROU with an IEEET1 exciter model. Suppose the detailed system consist of 3 buses with 2 transmission lines and 3 parallel RL loads. Also assume that the TS time step equals 100 EMT time steps. Assume that the external system is connected to the detailed system at 2 boundary buses.

The number of variables to be solved by TS3ph at each time step is
6000 (6 X 1000 buses) + 2700 (9 X 300 generators) = 8700
For each EMT time step the number of variables to be solved are
9 (3 X 3 buses) + 6 (3 X 2 lines) + 9 (3 X 3 loads) = 24
Since there are two boundary buses, the i_{bdry} vector is of size 6 (3 X 2 boundary buses).
Coupling 100 EMT time steps together the number of equations to be solved by this time-coupled EMT system is 100 X 24 + 100 X 6 = 3000
giving 8700 + 3000 = 11700 equations to be solved at each TS time step.

6.6 Proposed electromechanical and electromagnetic transients simulation strategy

The proposed implicitly coupled simulator, TSEMT, can by itself be used for a combined electromechanical and electromagnetic transients simulation. If the fast dynamics, harmonic voltages and currents, in the detailed system are of prime importance then the TSEMT simulator could be used for the entire simulation time length. Our interest in the TSEMT simulator is for analyzing the fast dynamics following disturbances only. Disturbances typically cause the generation of harmonic voltages and currents and the TSEMT simulator can be used only when harmonics are present. When there are no harmonics, a transient stability simulator is sufficient to simulate fundamental frequency, or relatively slow, dynamics and hence it should be used.

Hence the electromechanical and electromagnetic transients simulation strategy presented here is to use TSEMT selectively whenever there are harmonics and use TS3ph for the rest of the time frame. Such a strategy will be referred to as TS3ph-TSEMT for the remainder of this thesis. The TS3ph-TSEMT simulation strategy for a disturbance scenario, shown in Figure 6.9, is as follows:

1. TS3ph is run initially on the complete network during the pre-disturbance period.
2. At time t_1, a disturbance occurs and the complete network is split into a detailed system for EMT and an external system for TS3ph. The Thevenin equivalent for EMT is set up.
3. The combined set of TS3ph and coupled-in-time EMT equations for each TS3ph time step are solved using the proposed implicit coupled solution approach.
4. At time t_2, if the fast dynamics in the detailed system have died down then TSEMT is terminated, the network is merged again, and the relevant EMT variables are passed to the TS3ph simulator.
5. TS3ph is run on the entire network until end time.

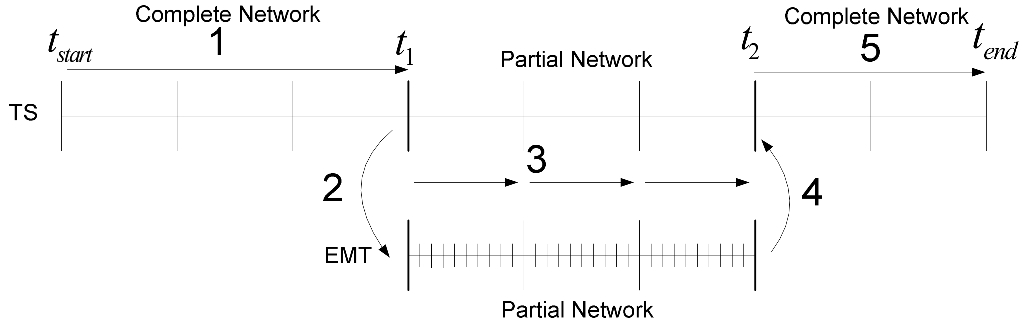


Figure 6.9. Combined TS3ph-TSEMT simulation strategy

6.6.1 Criterion for merging to TS3ph. Two schemes have been implemented in TS3ph-TSEMT to terminate TSEMT and merge to TS3ph.

6.6.1.1 Predefined merging time. This is a simple strategy for merging back to the TS3ph simulator. The implicitly coupled TSEMT simulator is terminated at a predefined time and passes the control to TS3ph. TS3ph is then continued for the remainder of the simulation.

6.6.1.2 Proposed merging criterion based on boundary bus voltage difference. The second merging strategy that has been implemented is based on the phasor voltage difference between the detailed and the external system boundary buses. The detailed system boundary bus phasor voltages are monitored to check if they are close enough to the external system boundary bus phasor voltages. If the difference is within an acceptable tolerance, then TSEMT is terminated and the control is passed to TS3ph. This strategy ensures that the non-fundamental frequency harmonics in the detailed system are negligible, so that TS3ph can be used on the entire system again. The criterion used for terminating TSEMT is as follows

$$||V_{bdry,EMT} - V_{bdry,TS}|| < \epsilon \tag{6.13}$$

where $V_{bdry,EMT}$ is the vector of detailed system boundary bus phasor voltages computed using Fourier analysis and $V_{bdry,TS}$ is the vector of the external system boundary bus voltage phasors. Furthermore, if the boundary bus voltage magnitudes are found to be low, such as during a fault, then the merging is avoided.

CHAPTER 7

TSEMT IMPLEMENTATION DETAILS AND SIMULATION RESULTS

This chapter explains the implementation details of the implicitly coupled TSEMT simulator such as steady state initialization, computation of external system equivalent for EMT, numerical integration scheme, etc. The simulation results of the TS3ph-TSEMT simulator are presented and compared with the TS3ph and EMT simulators. The experimental results of several reordering strategies and linear solution are finally discussed.

7.1 Steady state initialization

Similar to TS3ph, a Matlab code was written to do the TSEMT steady state initialization and save the needed data to files. The Matlab based package MATPOWER[79] was used for generating the power flow results. The Matlab code runs the power flow and generates ascii and binary files containing the data needed for the detailed system, external system and boundary. Currently, separate data files are used for the detailed and the external system. The Matlab code does the following preprocessing:

1. Run power flow using MATPOWER on the complete network to obtain steady state voltages.

2. Initialize the generator dynamic variables.

3. Read the external system generator, network and load data files.

4. Save the external system generator and load data in different ascii files.

5. Assemble the three phase Y matrix of the external system and save it in a binary file.

6. Save the initial conditions of the external system to an ascii file.

7. Create adjacency graph and vertex weights for the external system partititoning (only used for parallel runs).
8. Read the detailed system generator, network and load data files.
9. Save the external system generator and load data in different ascii files.
10. Assemble various matrices for EMT simulator and save them to ascii file.
11. Save the initial conditions of the detailed system to an ascii files.
12. Compute the network equivalents: Thevenin equivalent impedance and voltages and steady state boundary bus phasor current injection. Save Thevenin equivalent impedance to binary file.
13. Create the mapping of boundary buses between TS and EMT subsystems.
14. Initialize boundary currents i_{bdry0} for EMT.
15. Create the mapping of TS3ph subsystem network voltages in TSEMT to full TS3ph network voltages (for TS3ph-TSEMT).

The TSEMT C code then reads the different ascii and binary data files and sets up the TSEMT simulation.

7.2 Numerical integration scheme

Since the external system uses TS3ph, the default implicit trapezoidal integration scheme is used. Using this discretization, the algebraic equations to solve are as follows:

$$x(t + \Delta t) - x(t) - \frac{\Delta t}{2}(f(x(t + \Delta t)) + f(x(t))) = 0 \tag{7.1}$$

For EMT, a "θ integration" scheme is employed since it provides more flexibility to the choice of the numerical integration scheme.

$$x(t + \Delta t) - x(t) - (\theta f(x(t + \Delta t) + (1 - \theta)f(x(t))) = 0 \tag{7.2}$$

By varying the value of θ the numerical integration scheme can vary from Explicit Euler ($\theta = 0$) to Implicit Euler ($\theta = 1$). Note that $\theta = 0.5$ corresponds to the Implicit trapezoidal integration scheme. In the current TSEMT implementation, discretization of the EMT generator, network and load subsystem differential equations is done by implicit trapezoidal scheme ($\theta = 0.5$) while the boundary current equations are discretized using a hybrid implicit euler-trapezoidal scheme ($0.5 \leq \theta \leq 1$). The decision for using sucha scheme for the boundary current equations was done after observing that the boundary currents produce sustained (though bounded) numerical oscillations immediately following a disturbance if an implicit trapezoidal scheme is employed[51].

The EMTP program uses a scheme called critical damping adjustment (CDA) [51], [48] to alleviate the sustained numerical oscillations. In this scheme two half-steps of backward Euler are used immediately following a discontinuity and the integration scheme is switched back to implicit trapezoidal later on.

The CDA scheme could have been implemented for the TSEMT implementation but it was found that using a hybrid implicit euler-trapezoidal ($0.5 \leq \theta \leq 1$) scheme for the boundary current equations produces satisfactory results. Also since multiple EMT time steps are solved together, implementation of the CDA scheme would require additional steps and thus change the overall number of equations for the EMT subsystem. Hence the CDA scheme was not used.

7.3 Computing $v_{thev}(t)$

EMT has information about the thevenin equivalent voltage at its time boundary, i.e., at two consecutive TS time steps, so an instantaneous $v_{thev}(t)$ can be created directly. An important issue here is determining the instantaneous voltage $v_{thev}(t)$ for the EMT time steps that are not on the time time boundary, i.e., the EMT time steps in-between two consecutive TS time steps. The possible options for instantaneous $v_{thev}(t)$ for the in-between EMT time steps are

1. Compute $v_{thev}(t)$ from V_{thev} at the previous TS time step.

$$v_{thev}(t+i\Delta t_{EMT}) = |V_{thev}(t)|\sin(\omega(t+i\Delta t_{EMT}) + \angle V_{thev}(t)) \tag{7.3}$$

2. Compute $v_{thev}(t)$ from V_{thev} at the next TS time step.

$$v_{thev}(t+i\Delta t_{EMT}) = |V_{thev}(t+\Delta t_{TS})|\sin(\omega(t+i\Delta t_{EMT}) + \angle V_{thev}(t+\Delta t_{TS})) \tag{7.4}$$

3. Use $v_{thev}(t)$ computed from an interpolation of V_{thev} between the previous and next time step.

$$\begin{aligned}|V_{thev}(t+i\Delta t_{EMT})| &= |V_{thev}(t)| + i\Delta t_{EMT}\frac{|V_{thev}(t+\Delta t_{TS})| - |V_{thev}(t)|}{\Delta t_{TS}}\\ \angle V_{thev}(t+i\Delta t_{EMT}) &= V_{thev}(t+i\Delta t_{EMT}) + i\Delta t_{EMT}\frac{\angle V_{thev}(t+\Delta t_{TS}) - \angle V_{thev}(t)}{\Delta t_{TS}}\\ v_{thev}(t+i\Delta t_{EMT}) &= |V_{thev}(t+i\Delta t_{EMT})|\sin(\omega(t+i\Delta t_{EMT}) + \angle V_{thev}(t+i\Delta t_{EMT}))\end{aligned} \tag{7.5}$$

The existing hybrid simulators use option 1. Based on our simulation results as shown in 7.1, we found that option 2 is a better choice than 1 and 3, and we use option 2 for computing $v_{thev}(t)$ for TSEMT.

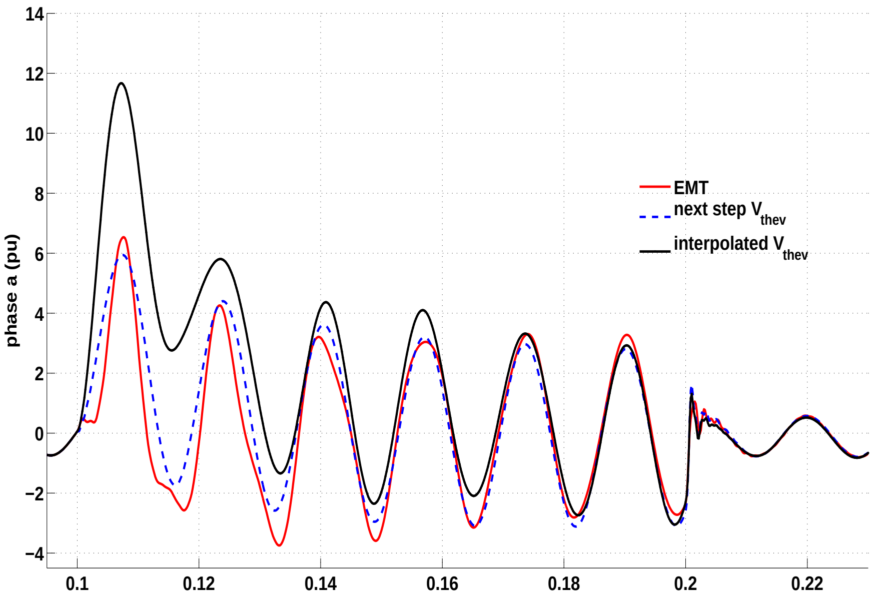


Figure 7.1. Comparison of different $v_{thev}(t)$ calculation

7.4 Disturbance Simulation

TS and EMT use different strategies to handle disturbances. TS uses an additional step t_f+ to simulate the disturbance on the TS network. During the t_f+ time step, only the algebraic equations for the network voltages and the generator stator currents are solved while the dynamic variables are kept fixed. The solution of only the algebraic equations follows the quasi steady state assumption used in TS, which implies that the network transients and stator electrical transients are much faster than the generator dynamics. Hence only the network and the stator algebraic equations need to be solved. It should be noted that while the generator dynamic variables remain fixed, their rate of change, or the derivatives, are updated due to the t_f+ network voltages and algebraic stator currents.

EMT, on the other hand, does not have any quasi steady state assumptions and no additional steps to handle disturbances are necessary.

To incorporate these different disturbance solution methods, the disturbance on/off time step simulation is modified for the implicitly coupled TSEMT simulator as shown in Figure 7.2

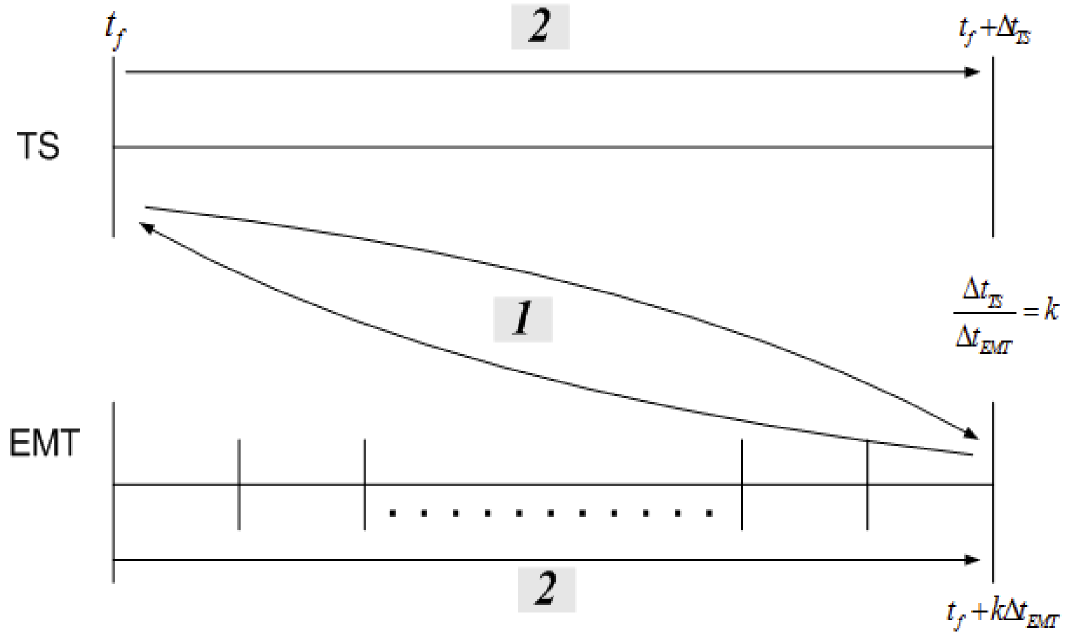


Figure 7.2. TSEMT disturbance time step

The disturbance solution process can be described as follows

1. At the fault time step t_f neither TS nor EMT have yet to apply the disturbance. The disturbance is now applied on the detailed system. The algebraic TS equations together with the time-coupled detailed system equations for one cycle are solved simultaneously. After the solution is obtained the derivatives for the dynamic TS variables are updated. A full period analysis after the fault is applied is necessary to accurately extract the fundamental frequency component of the interface variable I_{BDRY}.

2. The normal time-stepping of the TSEMT is continued with TS and coupled-in-time EMT equations solved simultaneously for the future time steps.

7.5 Simulation Results

The accuracy of the TS3ph-TSEMT simulator was tested on the WECC 9-bus and 118-bus systems. For all the simulations TS3ph uses a time step of one cycle, i.e., 0.016667 seconds and EMT uses a time step of one-hundredth of a cycle. This choice of TS3ph time step was used to ease the Fourier analysis process that requires a window of one cycle of data to compute phasor quantities.

For all the benchmarking simulations, the complete network is simulated by TS3ph initially (steady state). At a pre-specified pre-disturbance time (0.05 seconds chosen for the tests), the complete network is split into a detailed and external system and TSEMT simulator commences. The system is split before the disturbance time to ensure that the TSEMT simulator is initialized correctly and produces the same steady state values. (but perhaps this could be done after the fault-on time a, by backing up and initializing TSEMT). A three phase fault is applied in the interior of the detailed system at 0.1 seconds. The TSEMT simulator continues to execute till the merging algorithm, based on boundary bus voltage difference, detects that TSEMT can be terminated. On merging, the TSEMT simulator is terminated and the relevant variables from the external system and the detailed system are passed to TS3ph. TS3ph then continues to run until simulation end time.

7.5.1 9 bus: EMT subsystem 4-5-7. In this network split, the detailed system consists of the radial connection formed by buses 4,5,7 with two transmission lines 4-5 and 5-7 and a load at bus 8. All the generators are in the external system and modeled by a GENROU model with an IEEET1 exciter model. The loads in the TS and EMT regions are modeled as constant impedance loads. Figure 7.3 shows the division of the the detailed and the external system. The bounday consists of buses 4 and 7. A three phase balanced fault was applied in the interior of the detailed system at bus 5 at 0.1 seconds and its clearing time was varied.

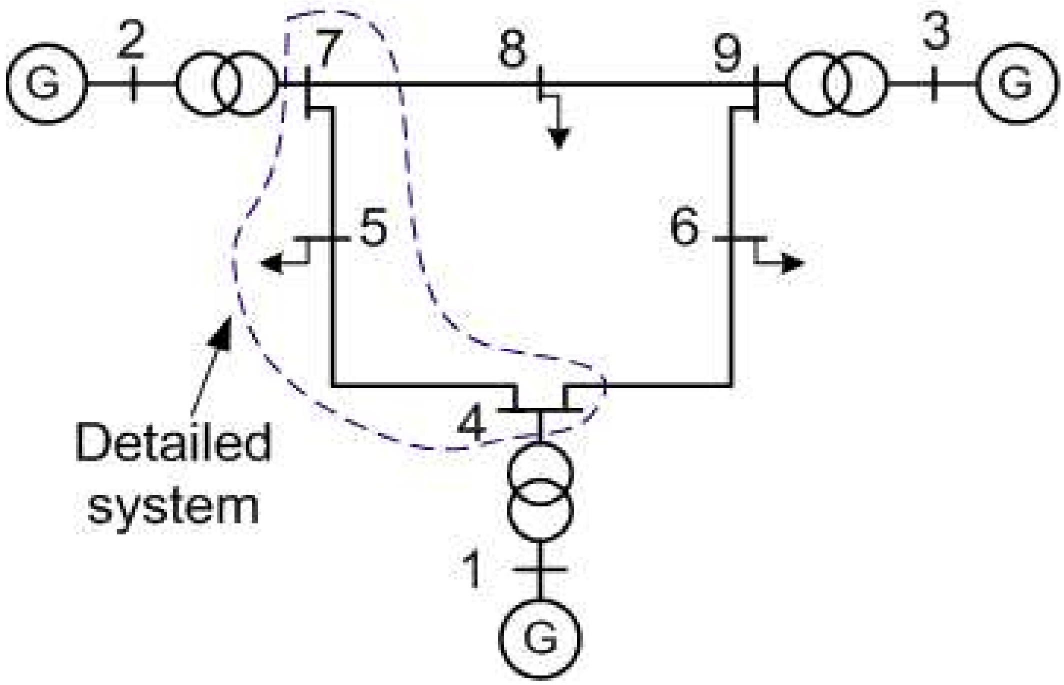


Figure 7.3. 9-bus system with buses 4,5,7 modeled in EMT

7.5.1.1 Stable case. For this scenario, a three phase fault in the detailed system was applied on bus 5 at 0.1 seconds and cleared at 0.2 seconds. TS3ph runs initially on the complete network till 0.05 seconds, at which time the system is split into an external and detailed system. The TSEMT simulator commences and runs past the fault clearing until 0.233 seconds, at which time the merging algorithm detects that the fundamental frequency phasor boundary voltages for TS and EMT are close enough to each other, and the system can be merged. The tolerance used for the merging algorithm was 0.01 pu. At 0.233 seconds, the TSEMT simulator is terminated and TS3ph continues for rest of the simulation period on the complete network.

Figure 7.4 shows the comparison of generator speeds for the TS3ph-TSEMT simulator with the full TS3ph and the EMT simulator. As seen there is a good agreement between the three simulators for the relatively slow generator speed dynamics for this stable system scenario.

Comparison of the three phase phasor voltages and the positive sequence phasor voltages for the boundary buses, 4 and 7, is shown in Figures 7.5 - 7.8. Following the three phase fault, the boundary bus voltages drop below 0.6 pu but eventually recover after the fault gets cleared at 0.2 seconds. The TSEMT simulator also shows good agreement for the boundary bus phasor voltages for the external system. Figures 7.9 and 7.10 show the instantaneous voltages for buses 4 and 7 when the TSEMT simulator is running. The fault causes high frequency harmonics in the boundary buses voltages. The harmonics in bus 4 voltages are more than bus 7 for this case. Qualitatively, the TSEMT simulator results are similar to the EMT simulator. The zoomed in plot of bus 4 voltage, figure 7.11, shows the actual differences. As seen, EMT can capture the high frequency components in the voltages while TSEMT can capture an approximate smoother version of the voltage waveform. The high frequency components are not captured by TSEMT since the external system equivalent contains only the fundamental frequency.

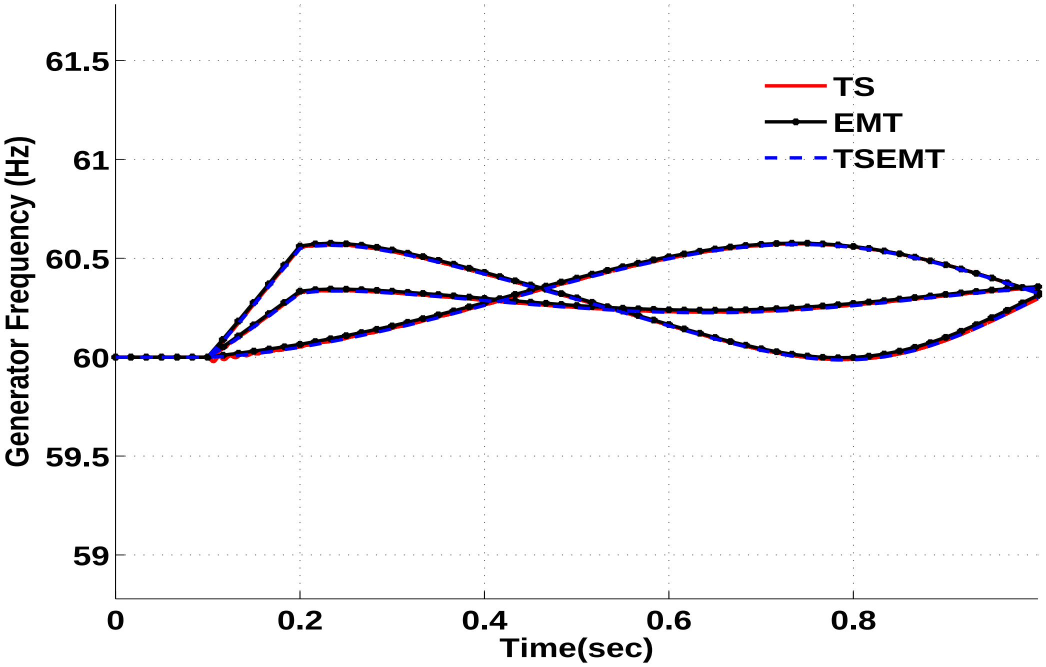


Figure 7.4. Generator frequency comparison

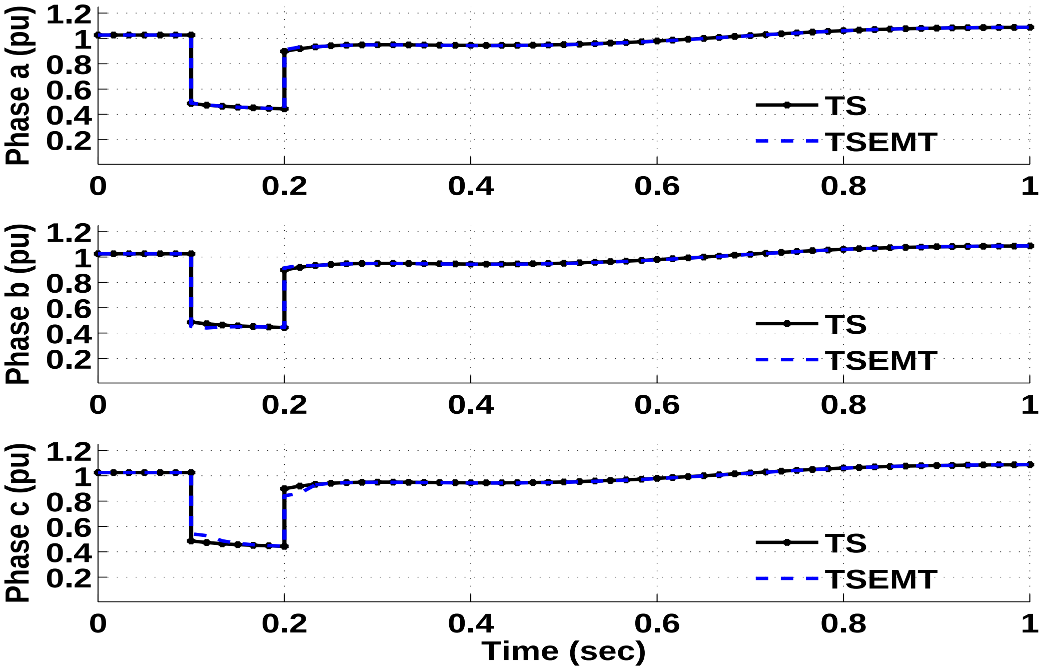


Figure 7.5. Three phase bus 4 phasor voltages

The boundary bus currents shown in Figures 7.12 - 7.13 show that the currents have a larger dc offset. The comparison of the TSEMT and full EMT boundary currents also show that there is more disparity between the two for the first few cycles following the fault. We believe that such a difference is due to the harmonics that cannot be correctly simulated by the fundamental frequency equivalent of the external system. As seen from Figure 7.11, the harmonics in the voltages damp out after 0.15 seconds. After 0.15 seconds, the boundary bus currents also show a good agreement. The zoomed-in plot of boundary bus currents, Figure 7.14, shows the actual difference between the TSEMT and full EMT in more detail.

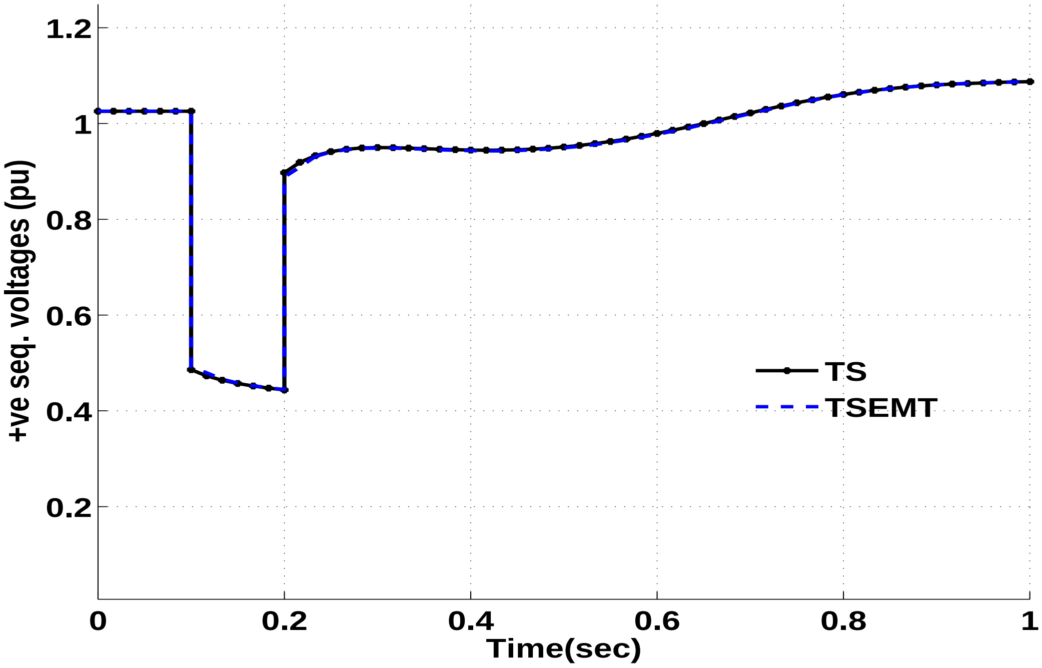


Figure 7.6. Positive sequence bus 4 phasor voltage

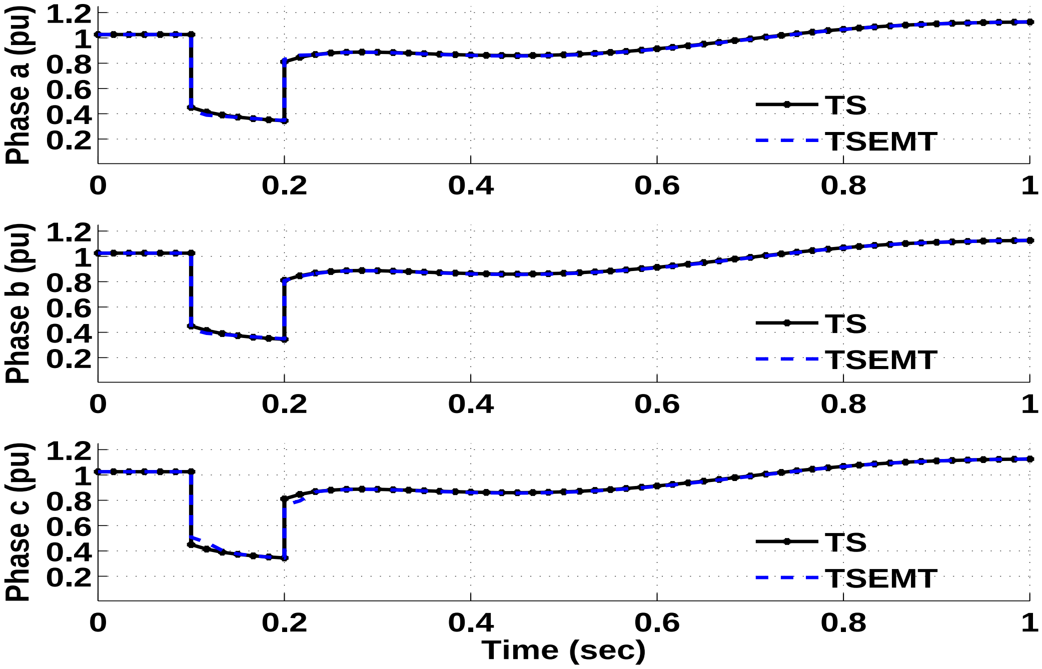


Figure 7.7. Three phase bus 7 phasor voltages

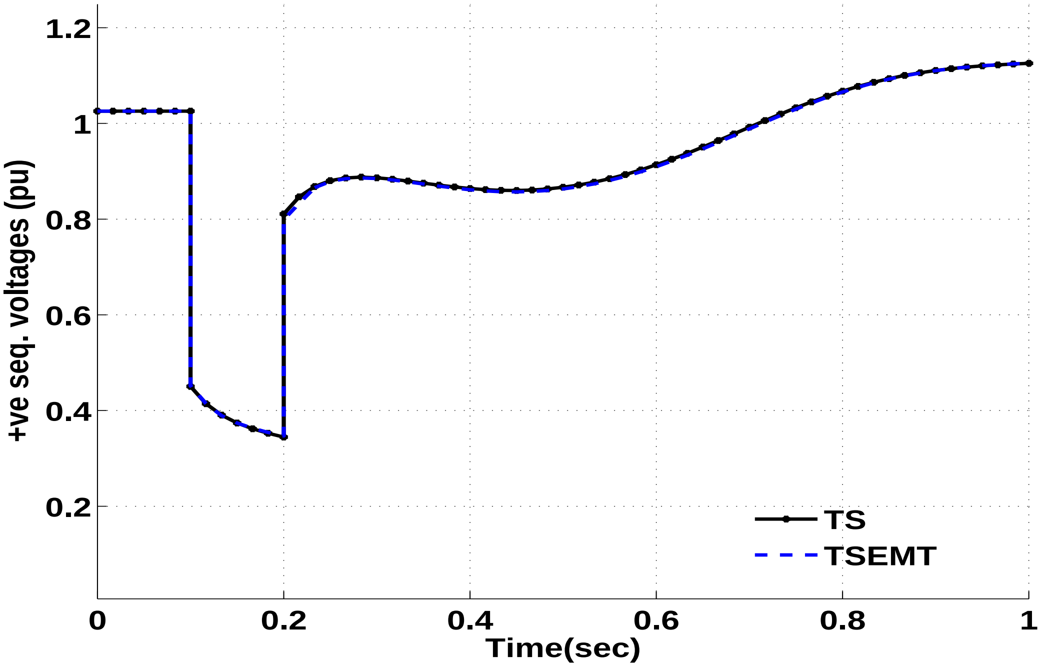


Figure 7.8. Positive sequence bus 7 phasor voltage

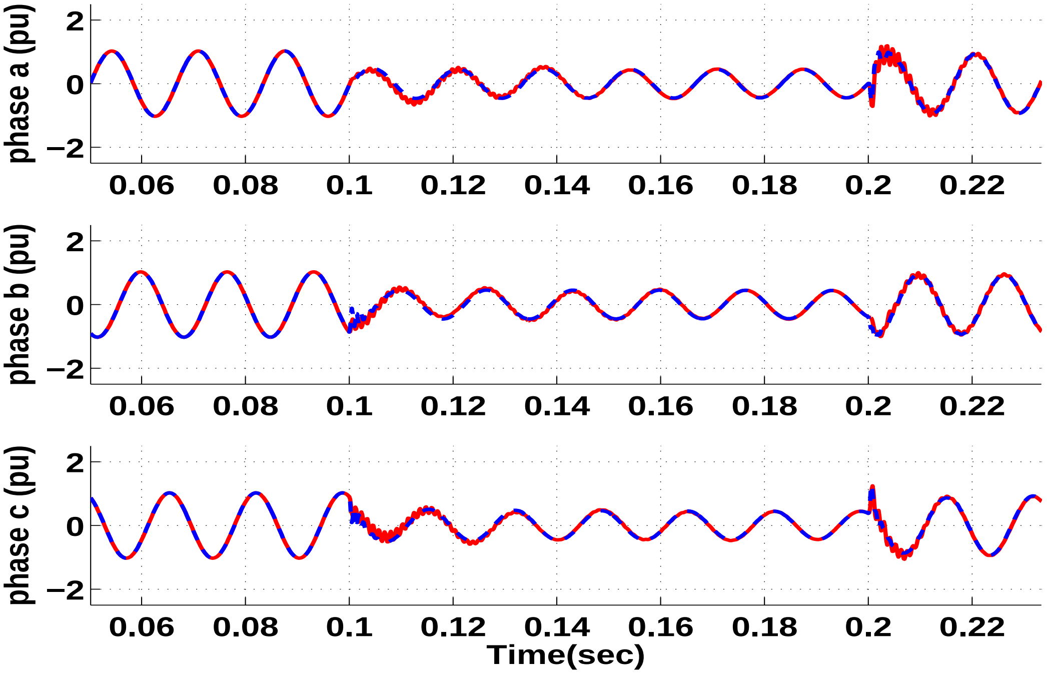


Figure 7.9. Boundary bus 4 instantaneous voltages with TSEMT (— EMT - - - TSEMT)

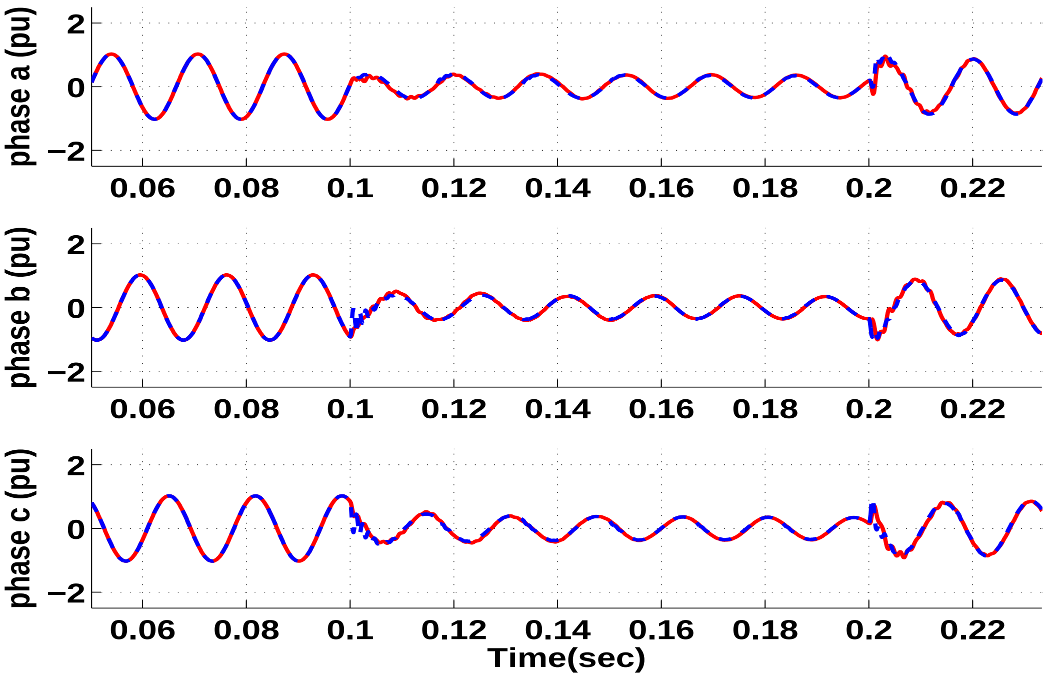


Figure 7.10. Boundary bus 7 instantaneous voltages with TSEMT (— EMT - - - TSEMT)

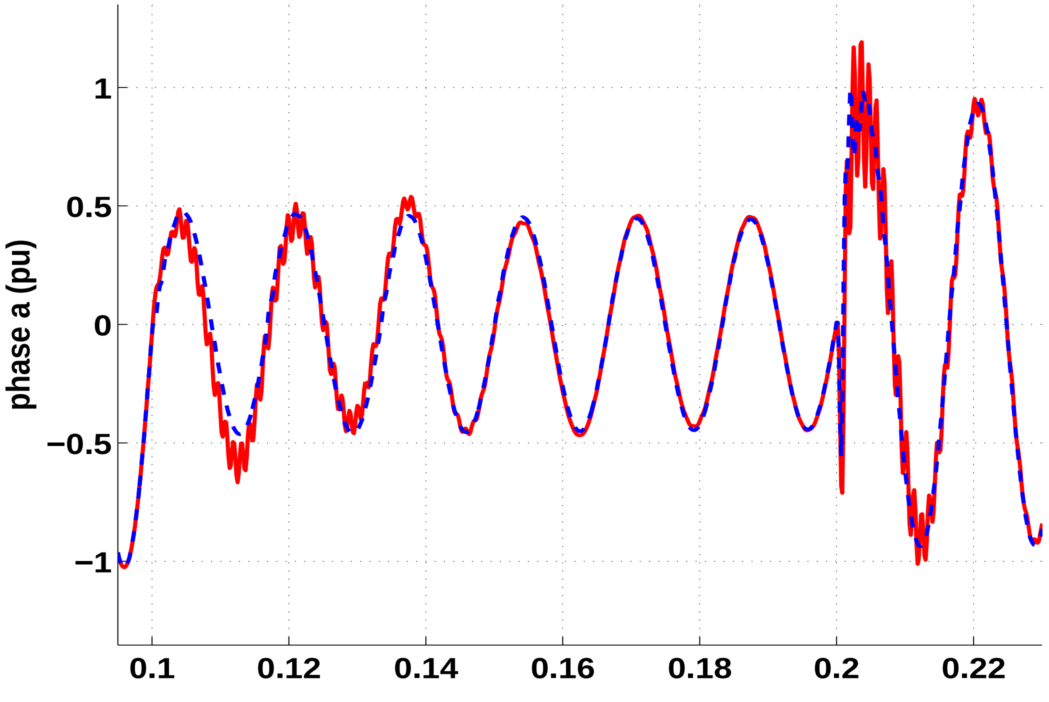


Figure 7.11. Zoomed in plot of bus 4 phase a voltage (— EMT - - - TSEMT)

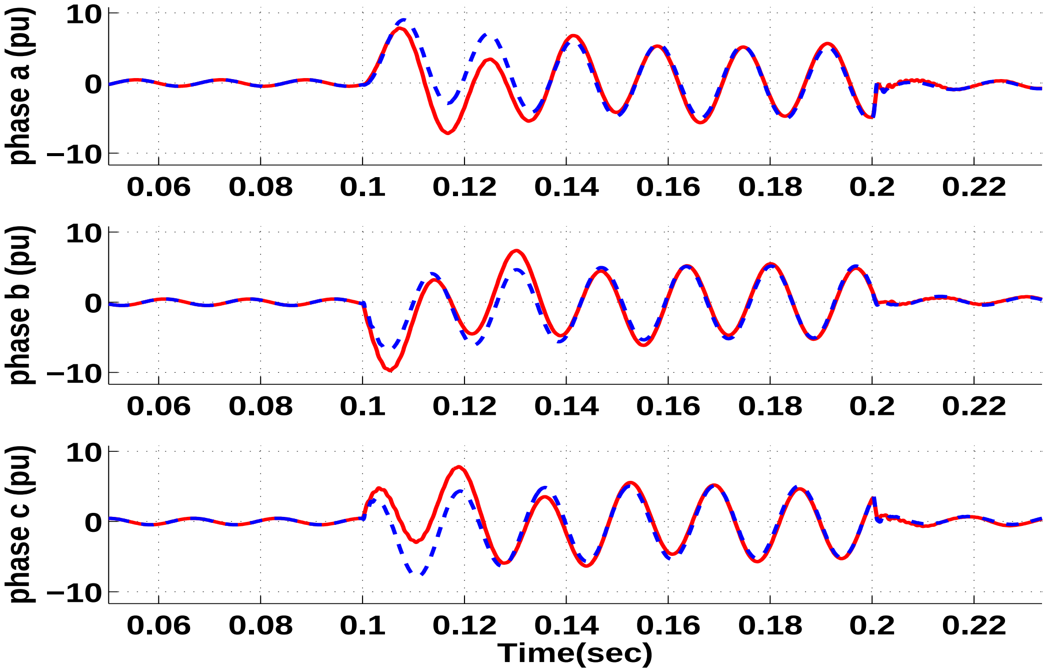


Figure 7.12. Bus 4 boundary currents with TSEMT (— EMT - - - TSEMT)

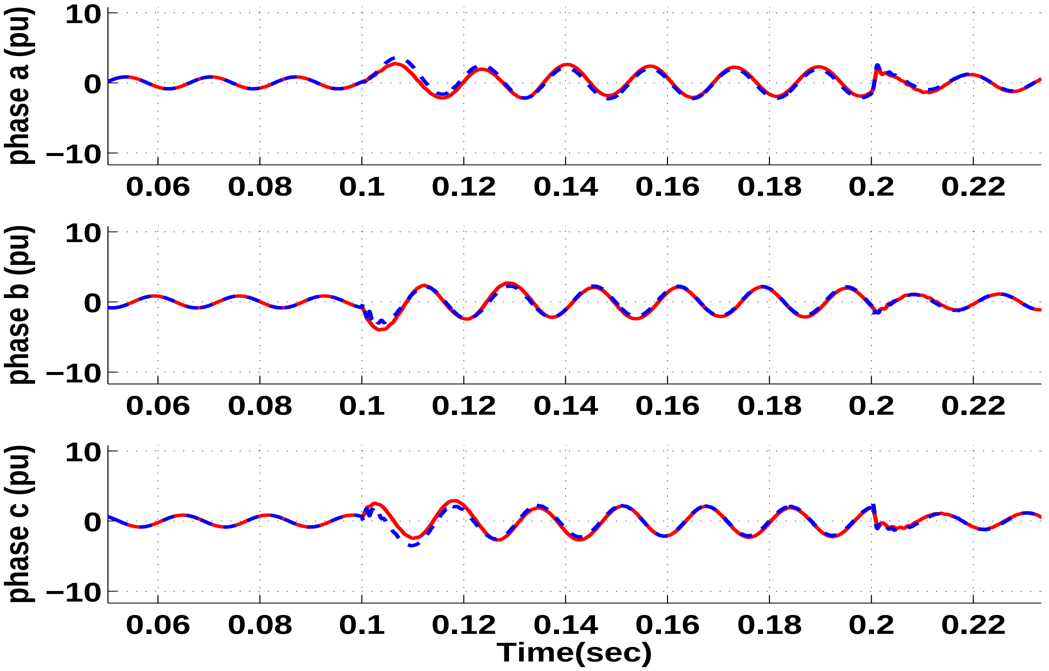


Figure 7.13. Bus 7 boundary currents with TSEMT (— EMT - - - TSEMT)

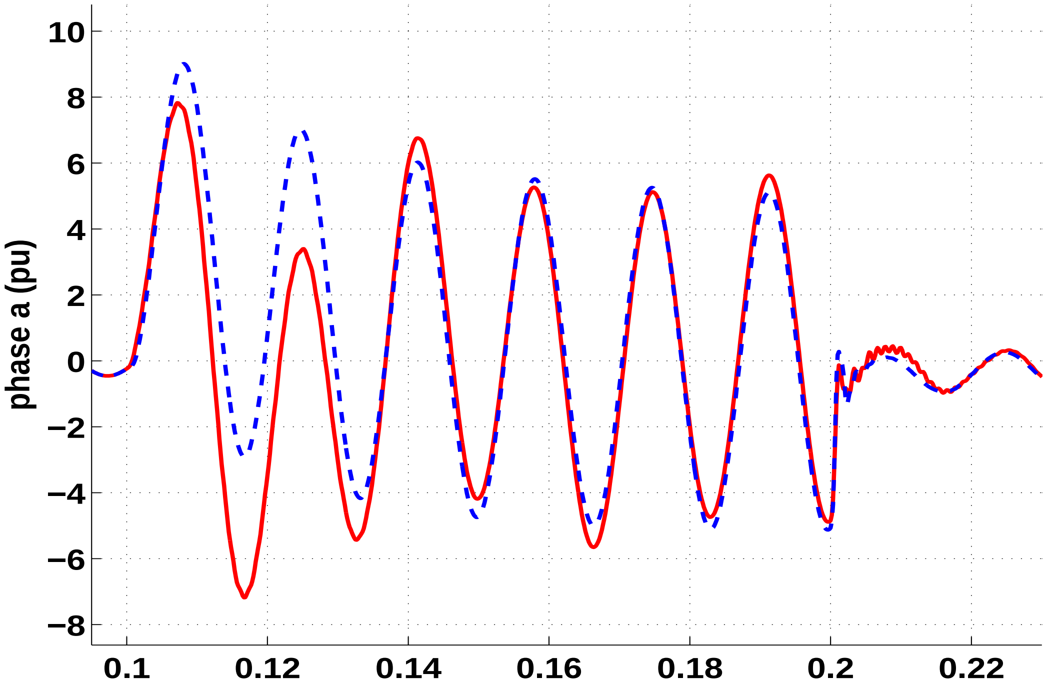


Figure 7.14. Zoomed in plot of bus 4 boundary current (— EMT - - - TSEMT)

7.5.1.2 Unstable case. It was found from earlier tests that for the 9-bus system a three phase fault for 12 cycles was enough to trigger the system to go unstable. For this test case, the three phase fault was applied at 0.1 seconds and cleared at 0.3 seconds. As before, TS3ph is run for the first 0.05 seconds following which the TSEMT simulator commences. At 0.33 seconds the TSEMT terminates as the boundary bus phasor voltages in the EMT and TS regions are equal. TS3ph continues from 0.33 seconds onwards.

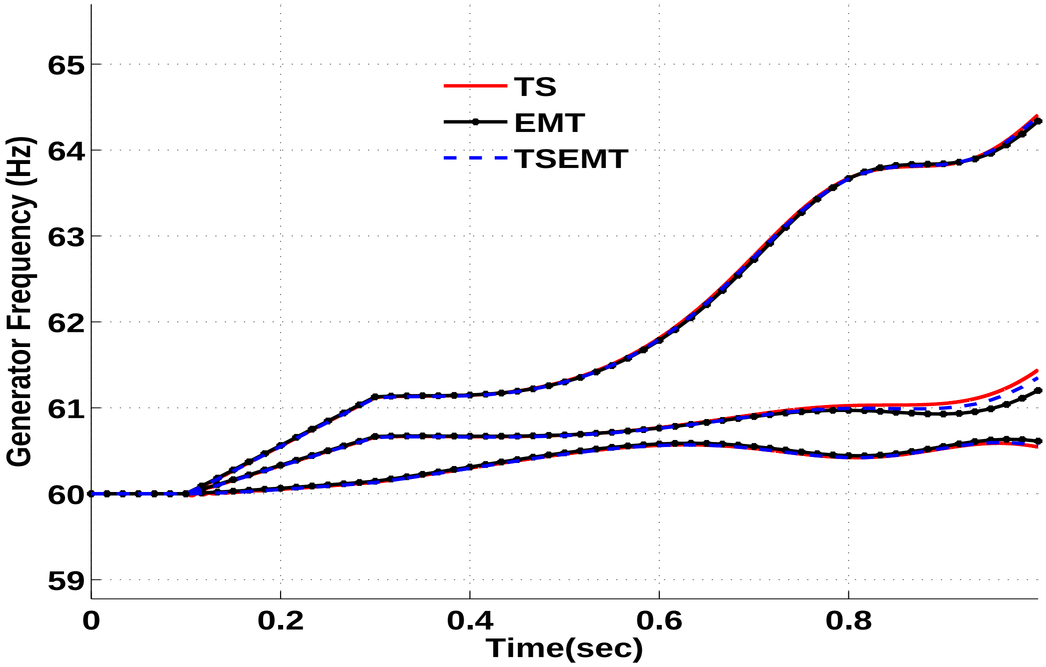


Figure 7.15. Generator frequency comparison for unstable case

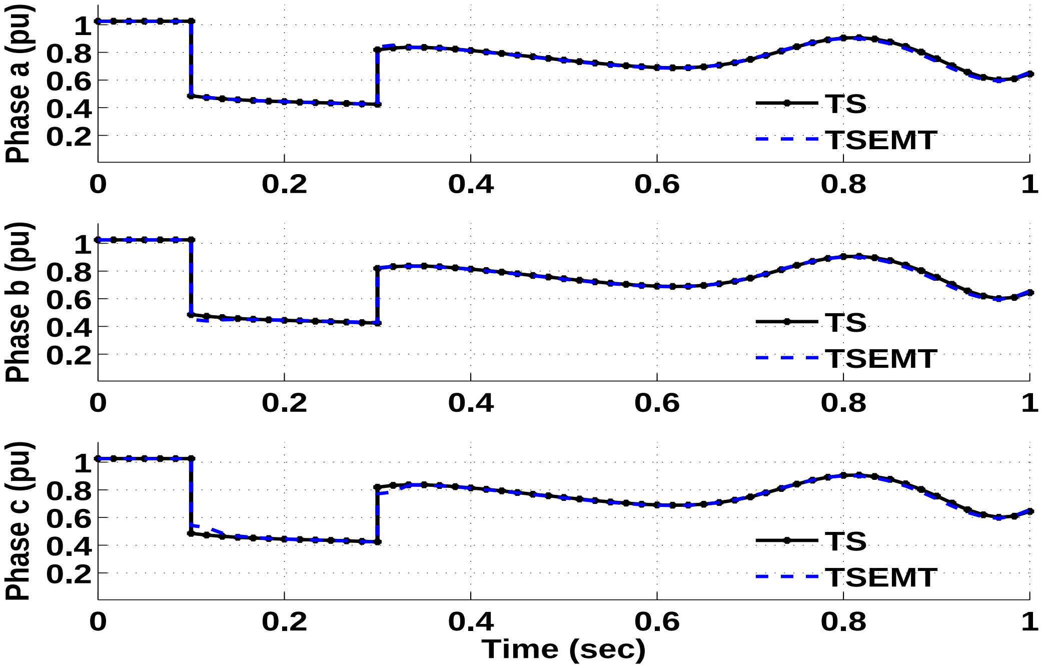


Figure 7.16. Three phase bus 4 phasor voltages for unstable case

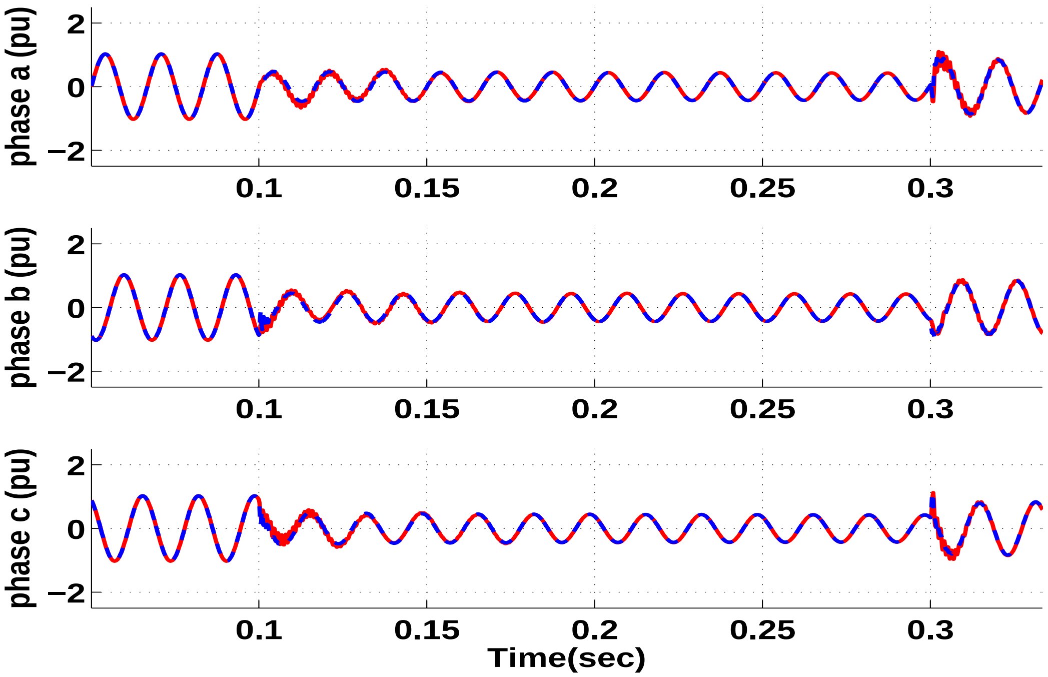


Figure 7.17. Boundary bus 4 instantaneous voltages with TSEMT (— EMT - - - TSEMT)

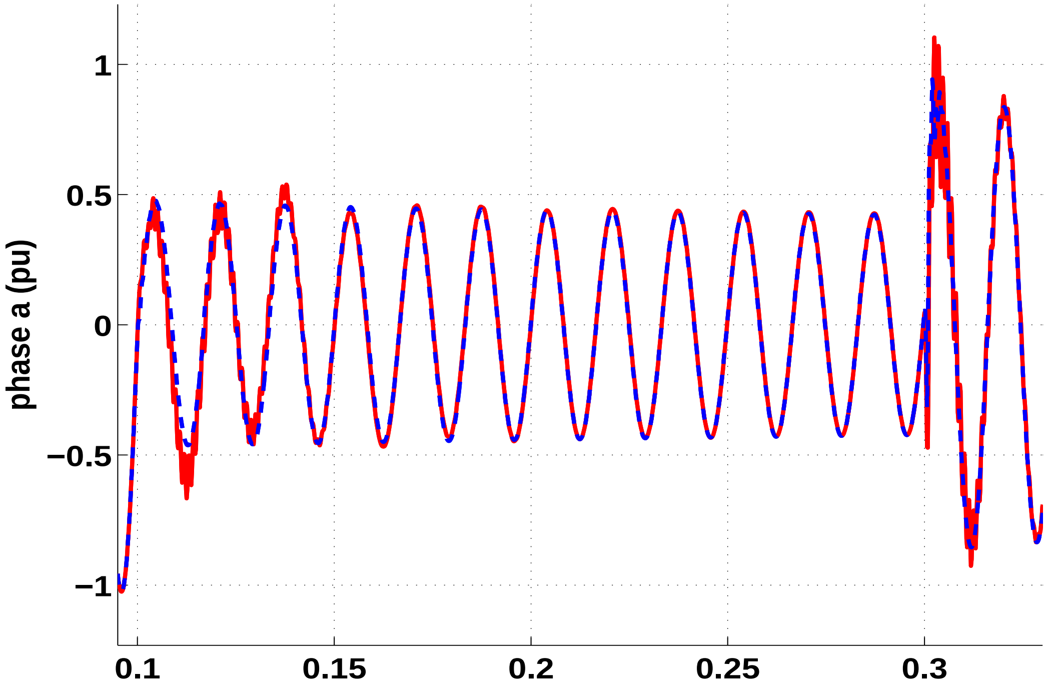


Figure 7.18. Zoomed in plot of bus 4 phase a voltage for unstable case (— EMT - - - TSEMT)

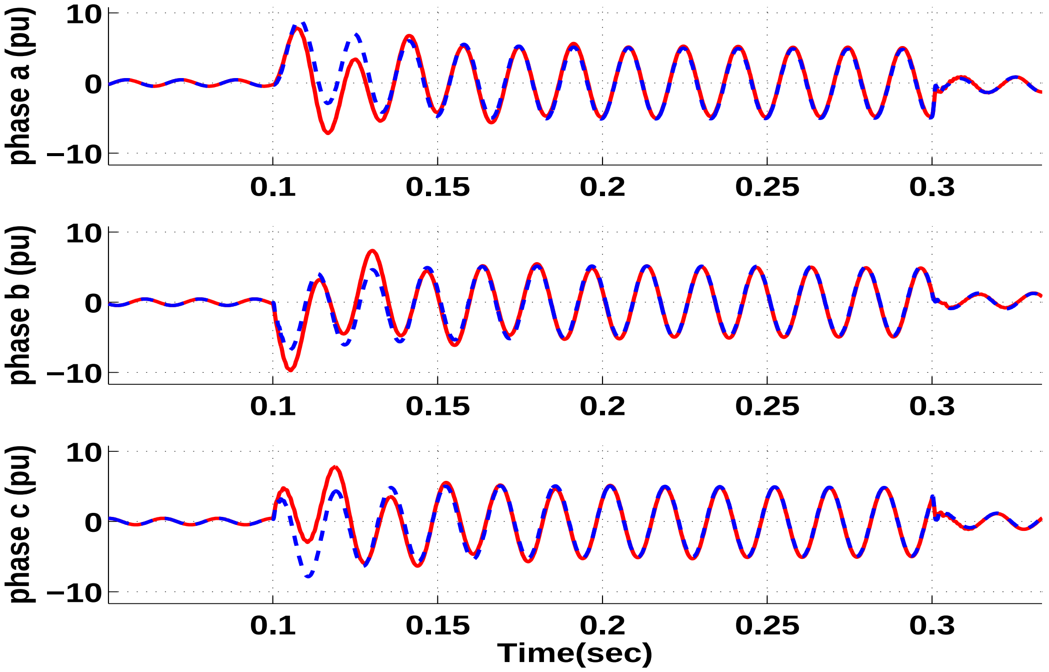


Figure 7.19. Bus 4 boundary currents with TSEMT (— EMT - - - TSEMT)

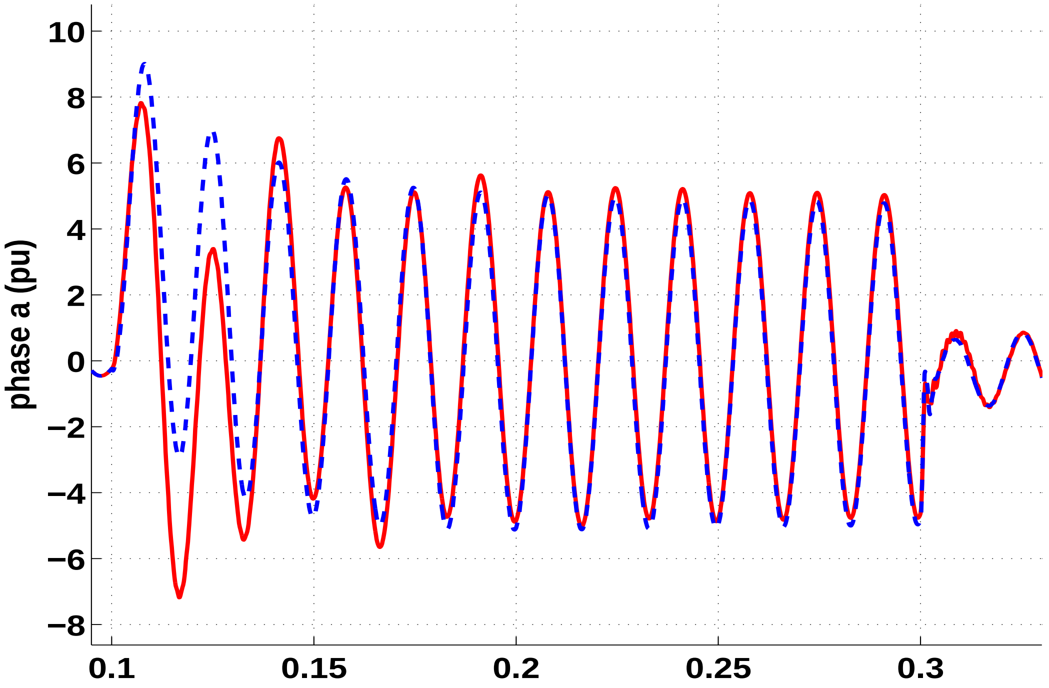


Figure 7.20. Zoomed in plot of bus 4 boundary current (— EMT - - - TSEMT)

7.5.2 9 bus : EMT subsystem 7-8-9. For this test case, the detailed system was moved to buses 7,8,9 with two transmission lines 7-8 and 8-9, with a load at bus 8. The generator and load modeling in the detailed and the external system is the same as in 7.5.1. A three phase fault is applied in the detailed system at bus 8 at 0.1 seconds and cleared at 0.2 seconds. The simulation results of the TS3ph-TSEMT simulator are shown in figures 7.22 - 7.30.

Again, the TS3ph-TSEMT simulator captures the slow dynamics of the external system to a good degree of agreement as seen in the plots of generator speeds and the boundary bus three phase and positive sequence voltages. After the fault, the higher frequency components are generated in the boundary bus instantaneous voltages. The TSEMT simulator can capture the wave shape to some approximately and not accurately as the external system equivalent is fundamental frequency based. The harmonics die out after 0.15 seconds and only fundamental frequency components exist in the instantaneous voltages.

The boundary bus currents experience a dc offset following the fault and there is disparity between the TSEMT and EMT results. The dc offset also diminishes after 0.15 seconds and the EMT and TSEMT boundary current waveforms show a good degree of agreement.

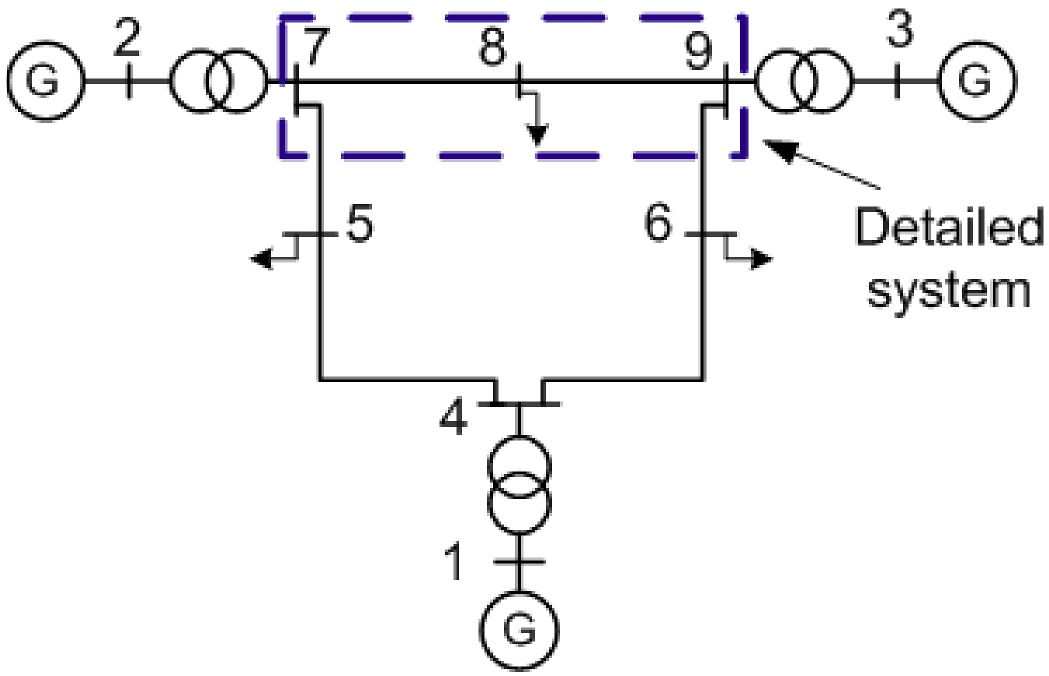


Figure 7.21. 9-bus system with buses 7,8,9 modeled in EMT

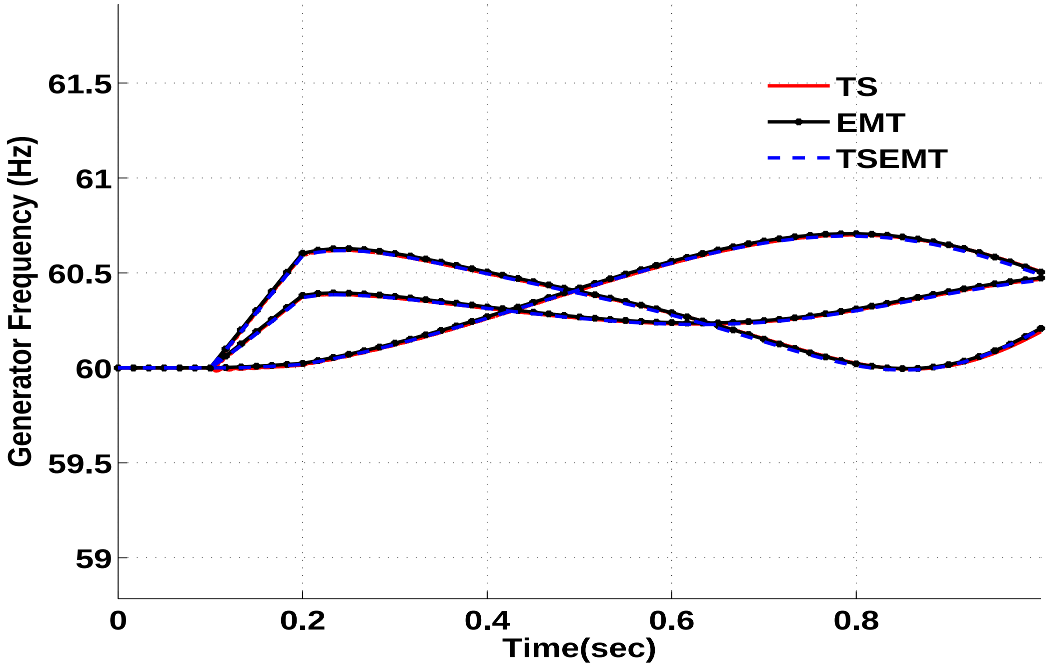


Figure 7.22. Generator frequency comparison with EMT subsystem 7-8-9

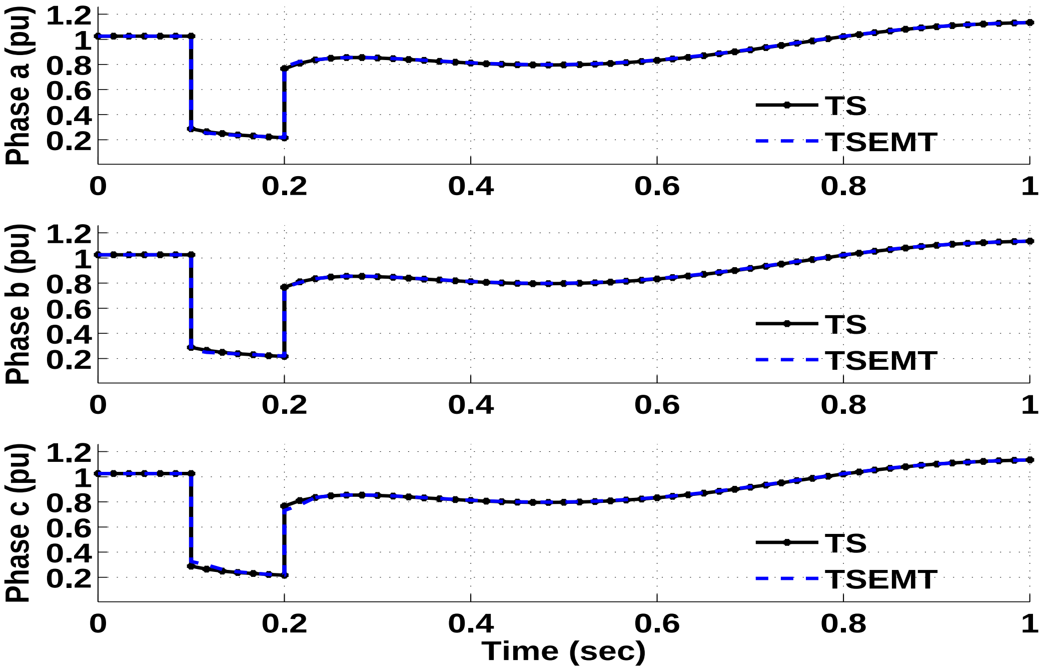


Figure 7.23. Three phase bus 7 phasor voltages

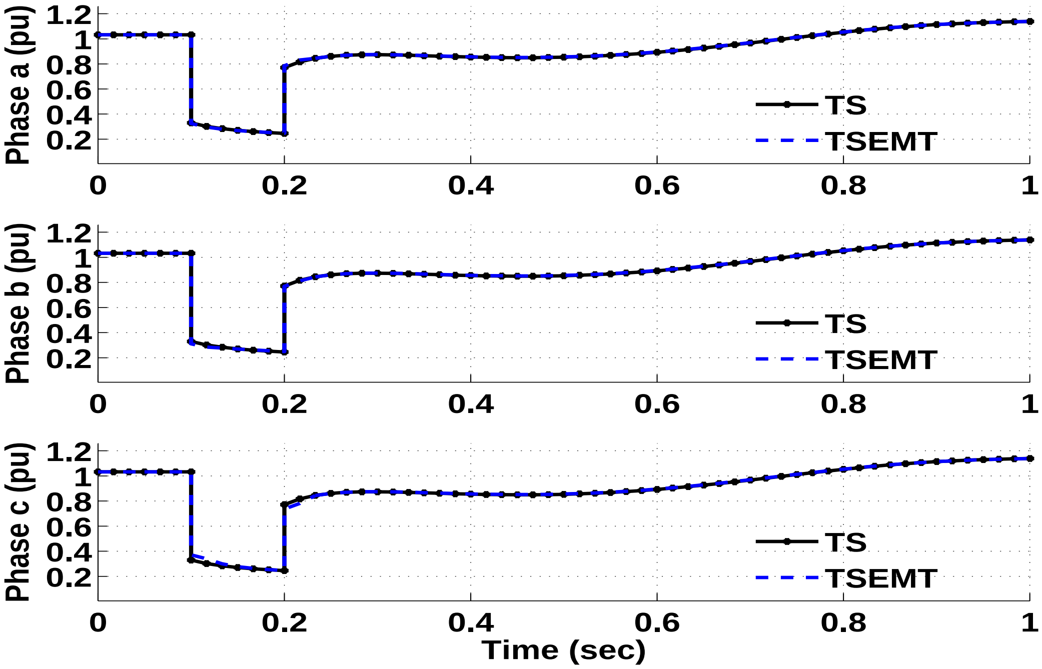


Figure 7.24. Three phase bus 9 phasor voltages

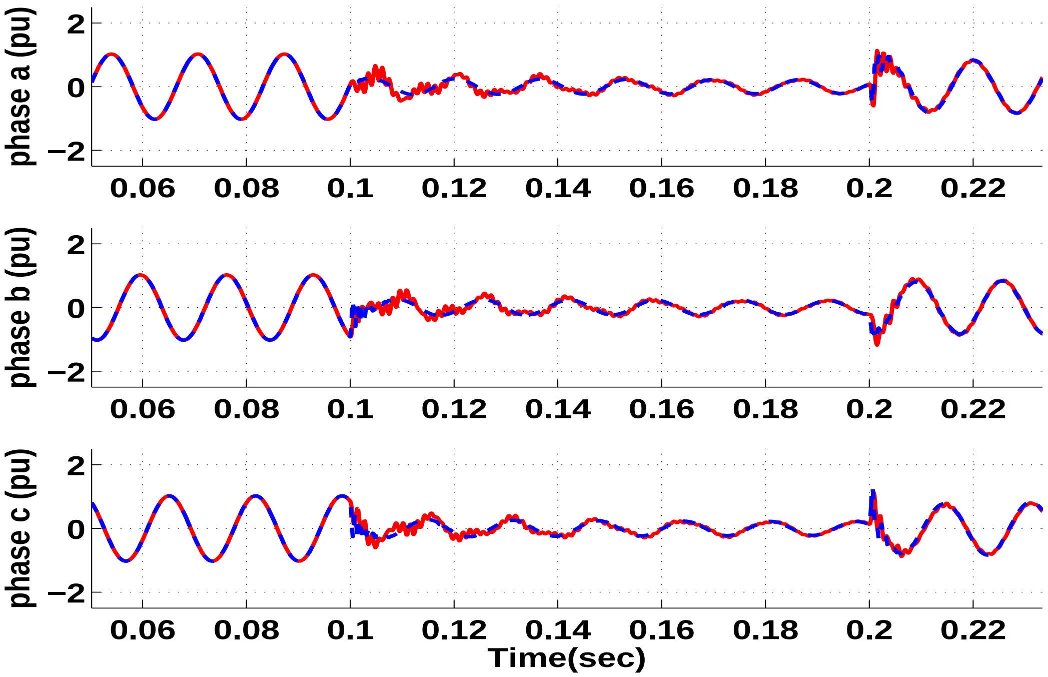


Figure 7.25. Boundary bus 7 instantaneous voltages with TSEMT (— EMT - - - TSEMT)

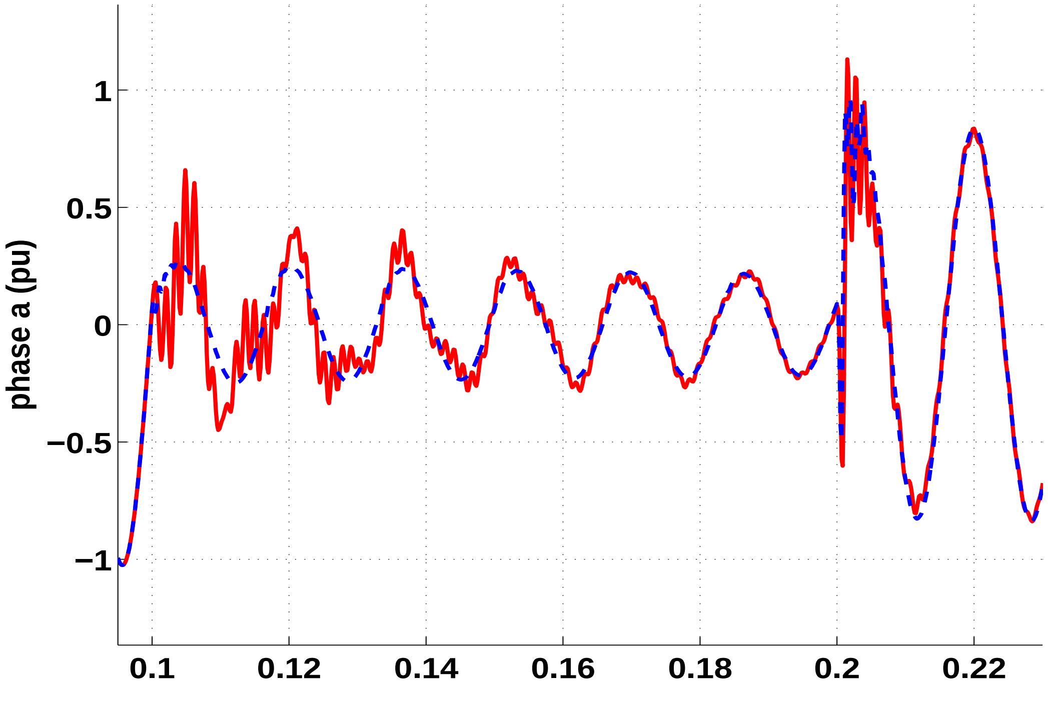


Figure 7.26. Zoomed in plot of bus 7 phase a voltage (— EMT - - - TSEMT)

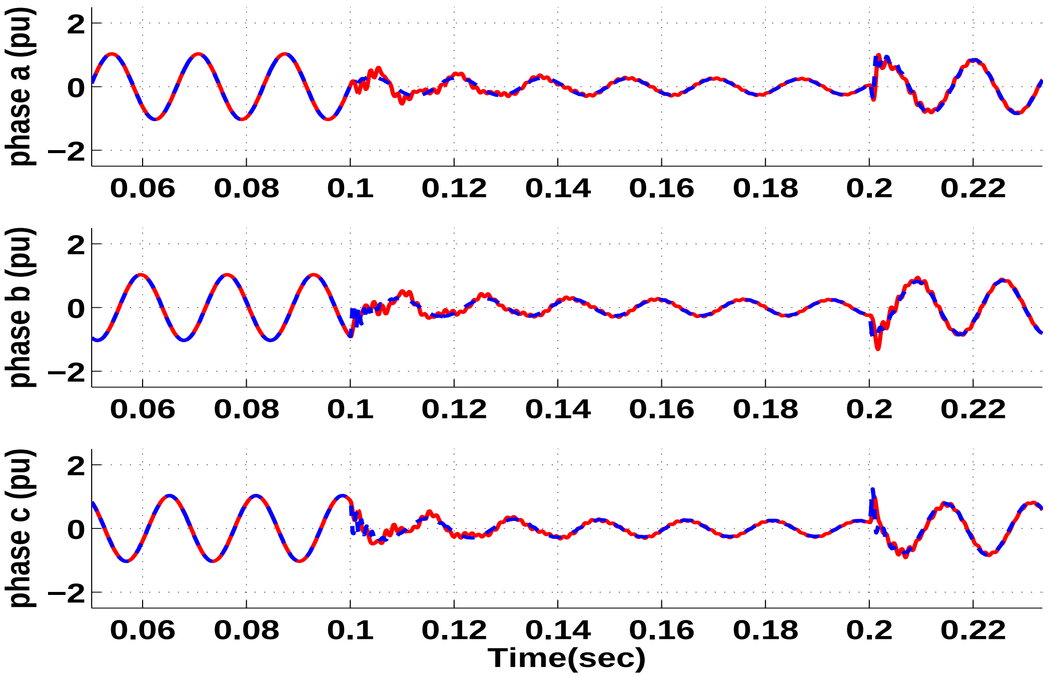


Figure 7.27. Boundary bus 9 instantaneous voltages with TSEMT (— EMT - - - TSEMT)

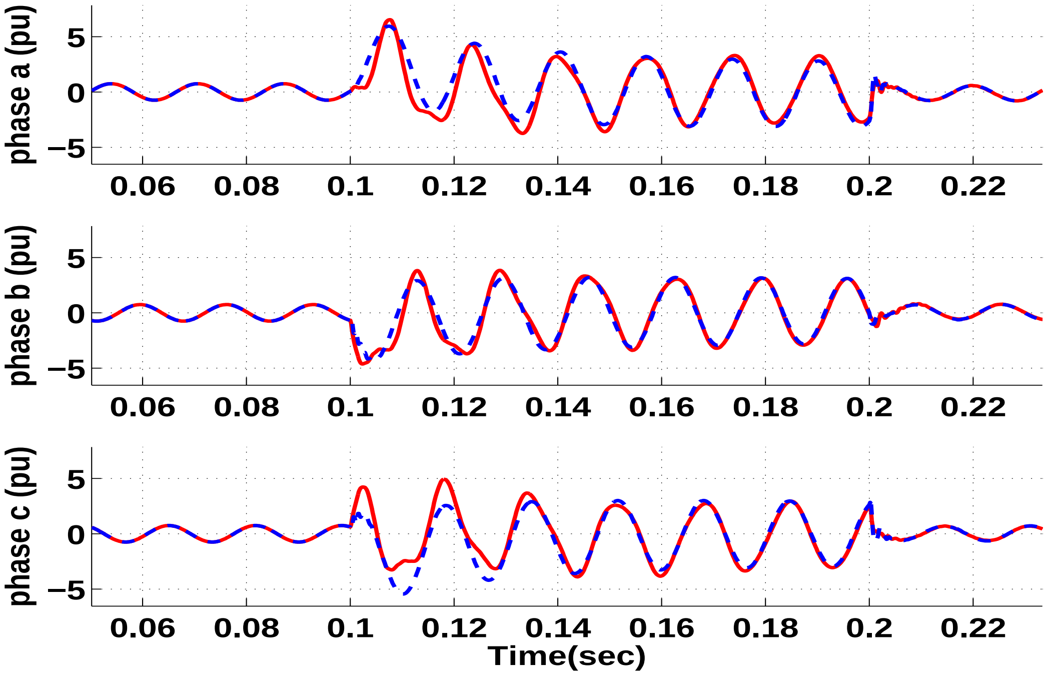


Figure 7.28. Bus 7 boundary currents with TSEMT (— EMT - - - TSEMT)

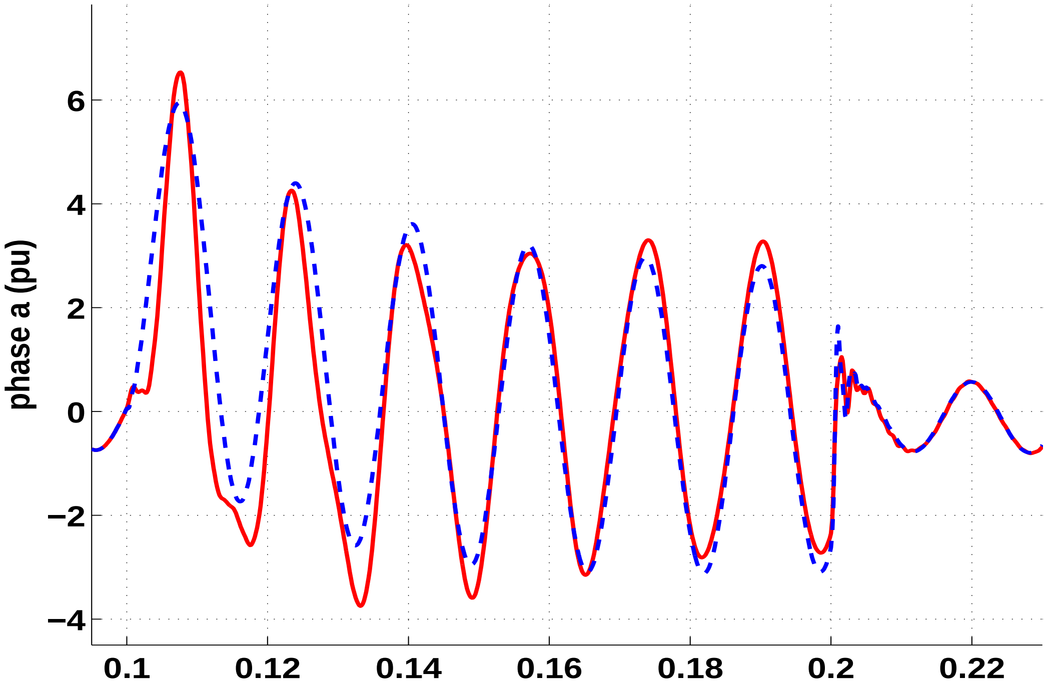


Figure 7.29. Zoomed in plot of bus 7 boundary current (— EMT - - - TSEMT)

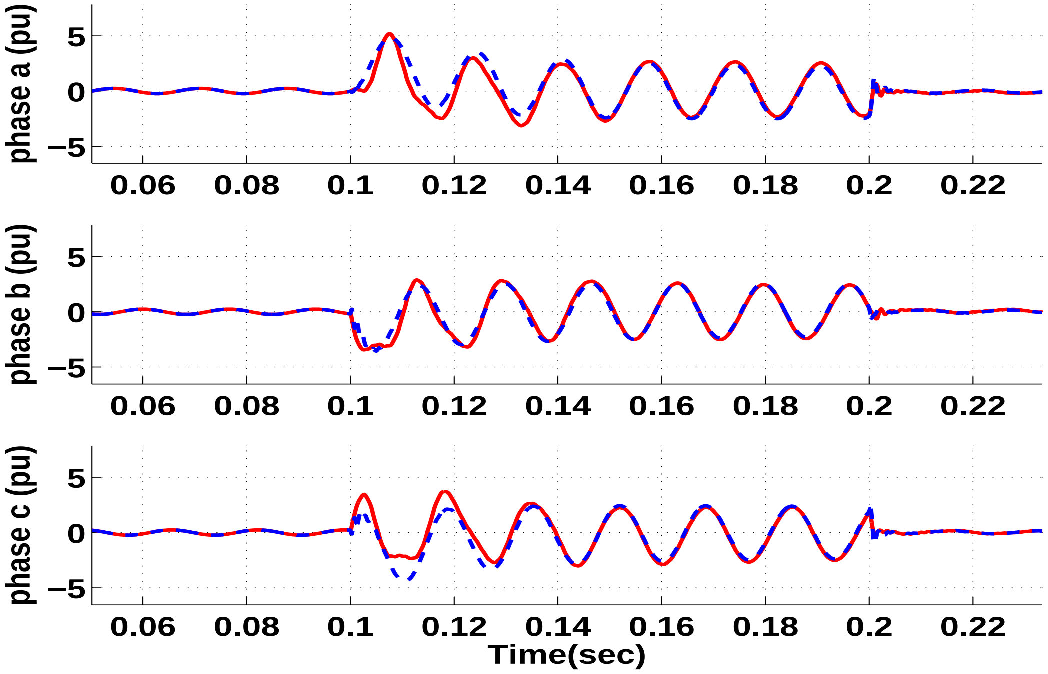


Figure 7.30. Bus 9 boundary currents with TSEMT (— EMT - - - TSEMT)

7.5.3 118 bus system. The TSEMT simulator was also tested on a larger 118 bus system with 186 lines. The detailed system was chosen as the radial connection formed by buses 20, 21, 22, and 23. This detailed system has a load at each bus and three transmission lines 20-21, 21-22, and 22-23. There are no generators in the detailed system. All the generators are in the external system and modeled as GENROU with an IEEE type 1 exciter model. All the loads, in the external and detailed system are modeled as constant impedances. The boundary buses for TS and EMT are buses 20 and 23.

As a disturbance scenario, a three phase fault was applied in the detailed system at bus 21 at 0.1 seconds and removed at 0.2 seconds. The TSEMT simulator commences from 0.05 seconds. At 0.2333 seconds, the merging algorithm detects that the phasor boundary bus voltages in the detailed and external system are within the specified tolerance of 0.01 pu and TSEMT is terminated. The control is passed back to TS3ph which runs until 1 second.

The simulation results for the 118 bus system are shown in Figures 7.31 - 7.37. The speeds of the generators closest to the detailed system are shown in Figure 7.31. Following the fault on bus 21, the generators closest to the detailed system experience a smaller speed deviation, maximum of 0.02 Hz, as compared to the WECC 9-bus system, which had a speed deviation of about 0.5 Hz for the stable case. It was verified by a running a 15 second simulation for this test case that the system regains stability.

the comparison of the phasor boundary bus voltages is shown in 7.32 and 7.33. Boundary bus 21, which is closer to the fault location than the other boundary bus, voltages drops below 0.4 pu following the fault and re-attain the steady state following the fault. The other boundary bus, bus 23, voltage drops around 0.88 pu during the fault-on period.

The boundary bus instantaneous voltages have less harmonic content immediately after the fault, as compared to the WECC 9-bus system which had more higher order frequency components in the voltages. This suggests that the TSEMT simulator should also be able to reproduce the full EMT boundary bus currents.

As seen from the boundary bus current plots, the TSEMT simulator is able to capture these boundary currents accurately immediately following the fault. This accuracy can be attributed to the presence of less harmonic content in the boundary bus voltages. Furthermore, we examined the voltage profile of the external system to determine if the external system is a 'strong', i.e., the external system voltages do not experience a large change following a fault in the detailed system. All the external system bus voltage profiles were flat suggesting that the external system chosen for this test case is a 'strong' external system. Perhaps, the serial interaction protocol could also produce similar results as the implicitly coupled TSEMT simulator, for this test scenario, since the external system is strong.

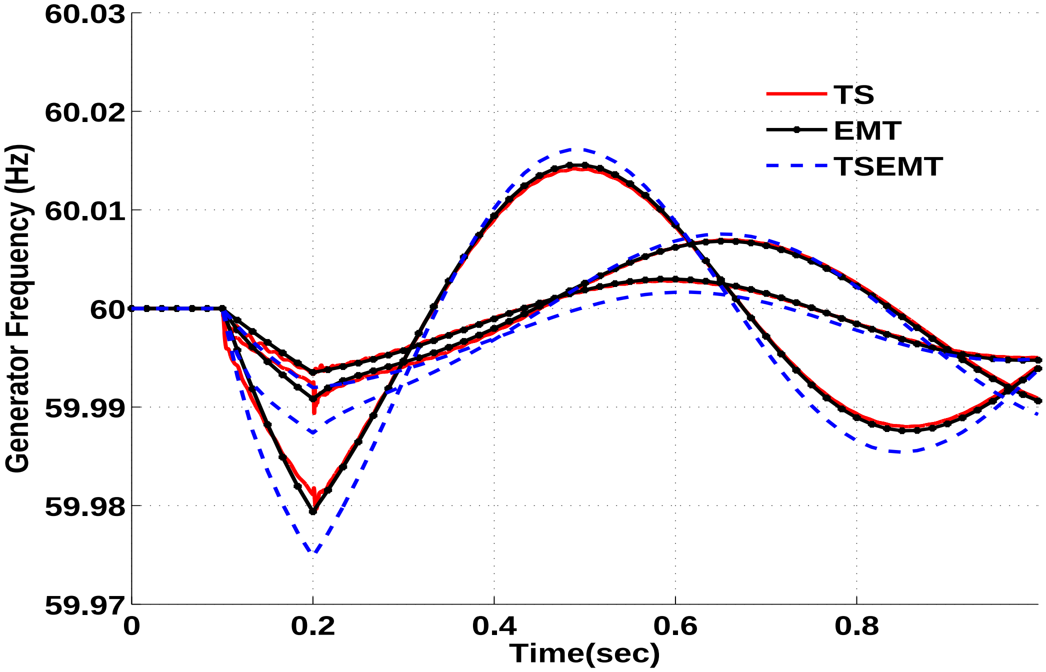


Figure 7.31. Frequencies for generators at buses 19, 24, and 25 in the external system

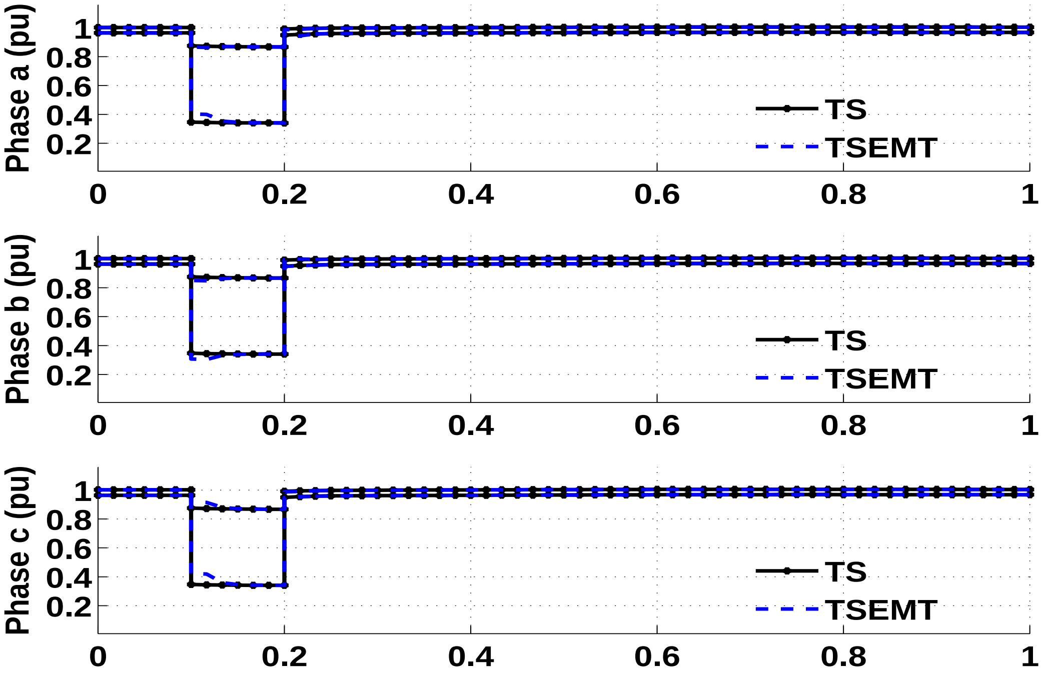


Figure 7.32. Three phase phasor voltages for boundary buses 20 and 23

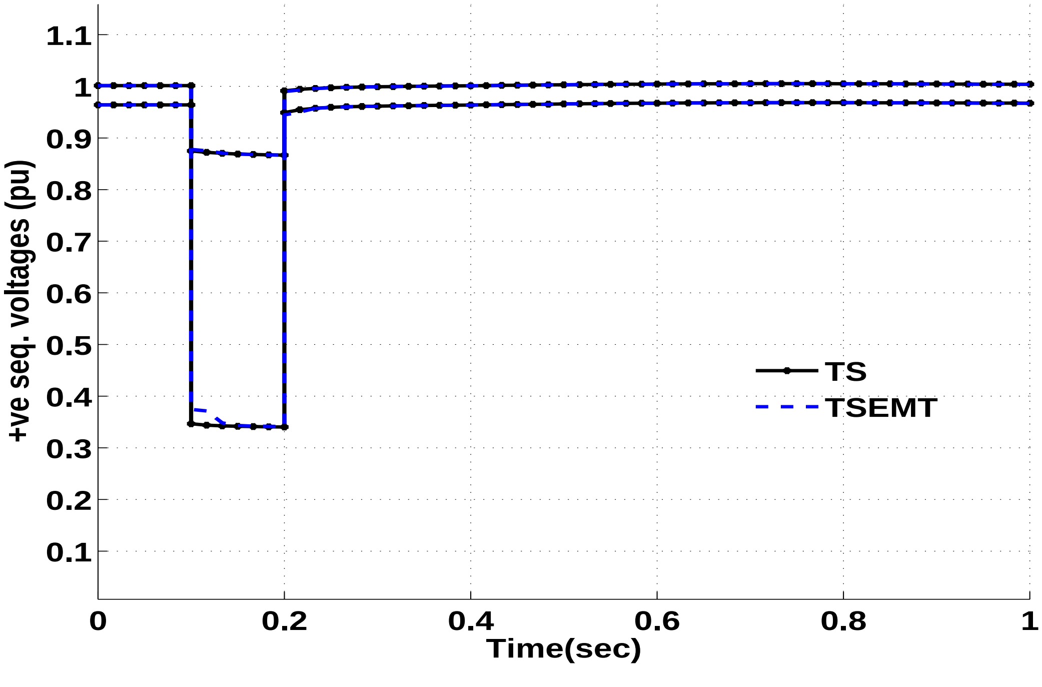


Figure 7.33. Positive sequence phasor voltages for boundary buses 20 and 23

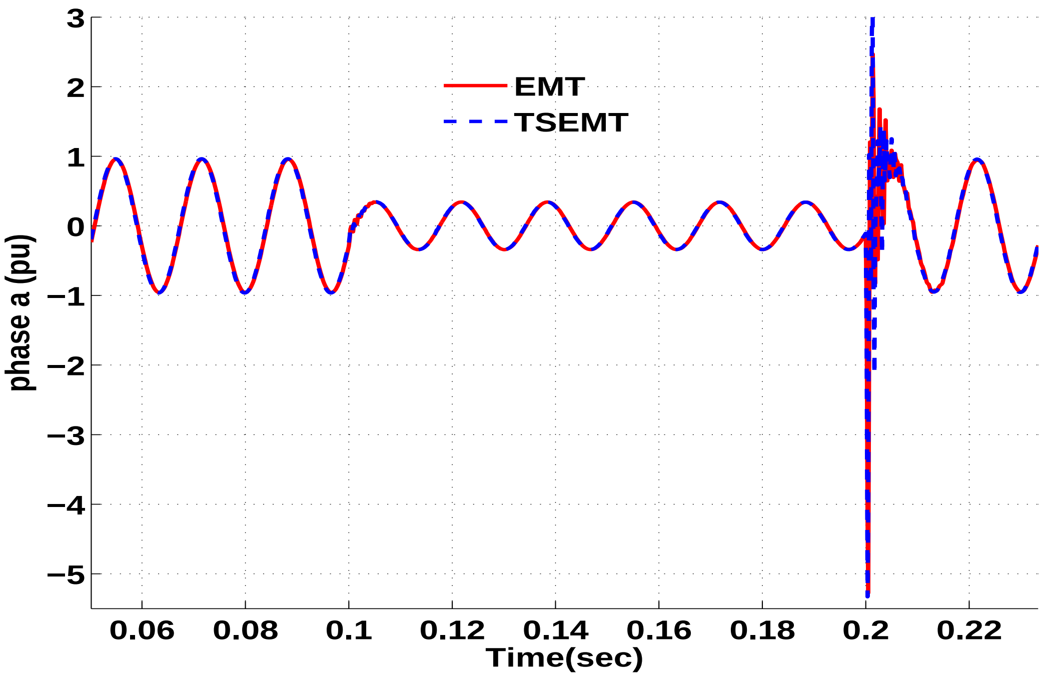


Figure 7.34. Bus 20 phase a instantaneous voltages (— EMT - - - TSEMT)

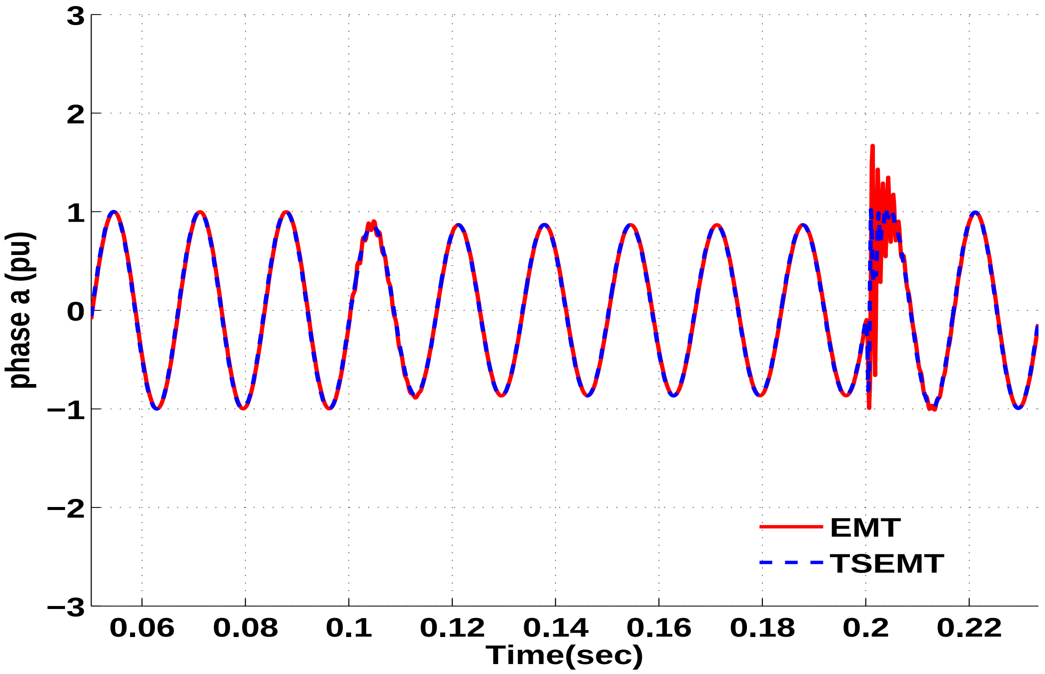


Figure 7.35. Bus 23 phase a instantaneous voltages (— EMT - - - TSEMT)

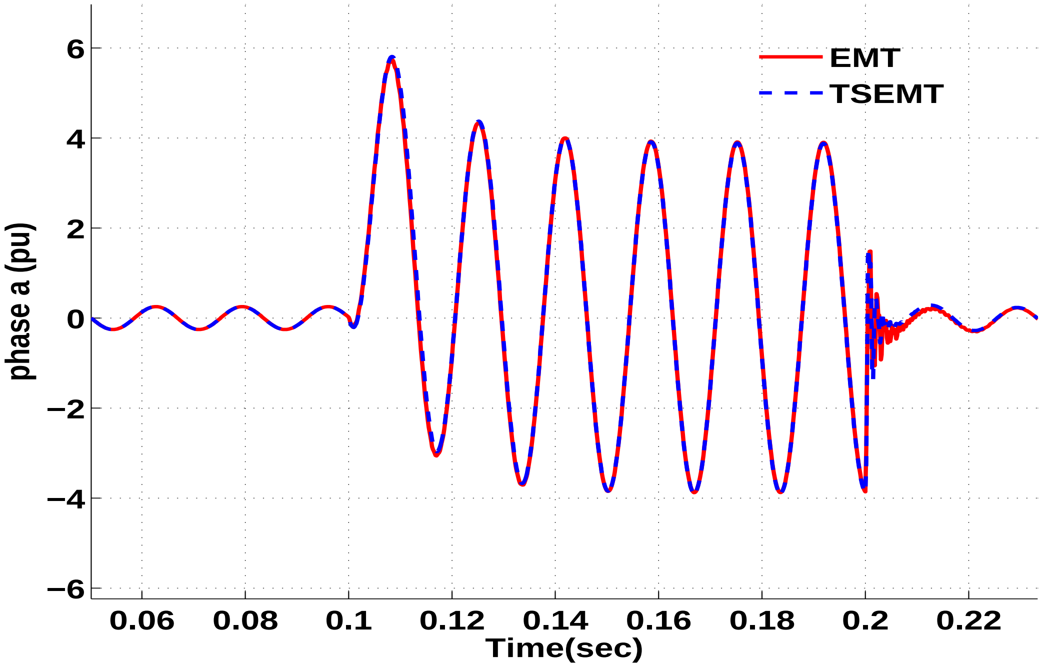


Figure 7.36. Bus 20 phase a instantaneous boundary current (— EMT - - - TSEMT)

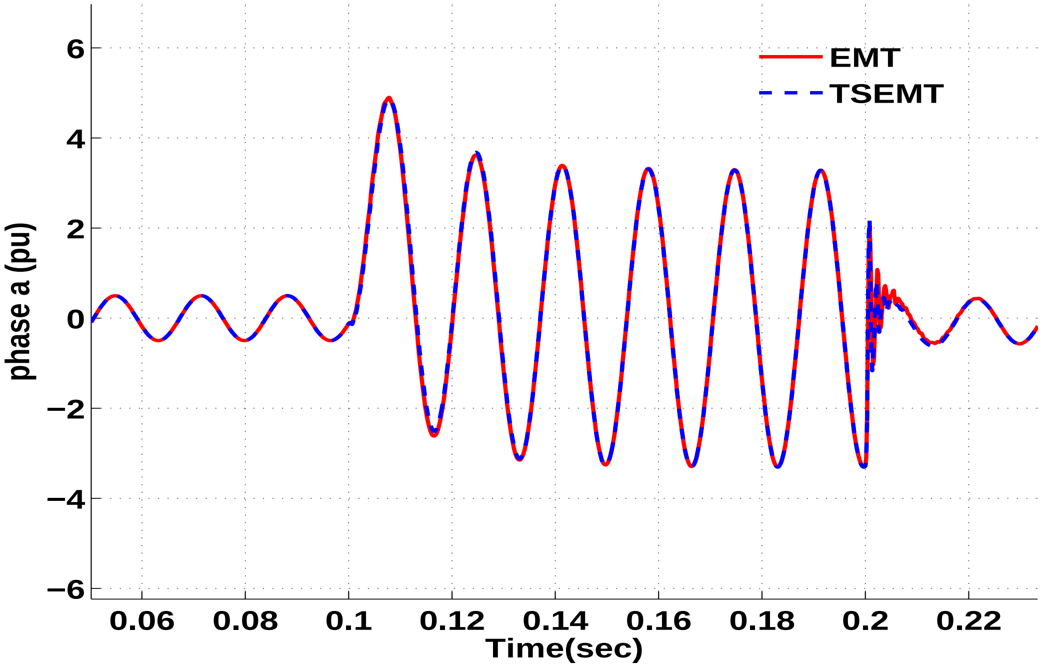


Figure 7.37. Bus 23 phase a instantaneous boundary current (— EMT - - - TSEMT)

7.6 Optimizing sequential TSEMT and TS3ph-TSEMT code

As in the case of TS3ph, we experimented with various reordering strategies and preconditioners to tune the linear solver for best performance. Only iterative linear solver GMRES, with different preconditioners, was tested for TSEMT since for TS3ph the iterative solver GMRES had better performance than the direct method. The code was compiled with optimization level -O3 and tested on an Intel Xeon 2.53 GHz processor, 12 GB RAM, 4 cores and 8 MB cache. We tested three test systems: 9 bus, 118 bus, and 1180 bus and present the different reordering and preconditioning results in the following subsection. The description of the 118 bus and 1180 bus systems is given in Appendix A.

7.6.1 Various reordering strategies.

Table 7.1. Reordering scheme non-zeros for the TSEMT 9 bus system

Ordering Type	Jacobian Matrix nnz	Factored Matrix nnz	Factor Fill Ratio
natural	32103	1083720	33.7576
nd	32103	240509	7.49179
rcm	32103	1110031	34.5772
1wd	32103	1103452	34.3722
qmd	**32103**	**197577**	**6.15447**

Table 7.2. Reordering scheme non-zeros for the TSEMT 118 bus system

Ordering Type	Jacobian Matrix nnz	Factored Matrix nnz	Factor Fill Ratio
natural	63276	2353524	37.1964
nd	**63276**	**702027**	**11.0947**
rcm	63276	1657228	26.1905
1wd	63276	1647163	26.0314
qmd	63276	794745	12.56

As in TS3ph, the *qmd* ordering seems to be the best for these three test systems. For the 118 bus system, *nd* strategy is the best but *qmd* is not far behind.

Table 7.3. Reordering scheme non-zeros for the TSEMT 1180 bus system

Ordering Type	Jacobian Matrix nnz	Factored Matrix nnz	Factor Fill Ratio
natural	271464	15573264	57.3677
nd	271464	6467667	23.8251
rcm	271464	5800012	21.3657
1wd	271464	6302770	23.2177
qmd	**271464**	**1739715**	**6.40864**

7.6.2 Optimizing linear solves.

For TSEMT, we experimented with exact and level-based incomplete factorization, ILU(k), preconditioners and also experimented with PETSc's multiphysics preconditioners. The multiphysics preconditioner class, called *fieldsplit*, is suitable for preconditioning coupled multiphysics problems. The *fieldsplit* preconditioner allows the use of custom linear solver for each physics domain along with its own preconditioner, reordering strategy and all the other intricacies.

Our interest in using the multiphysics preconditioners arises from the structure of the Jacobian matrix for TSEMT. Figure 7.38 shows the TSEMT Jacobian structure for the 118 bus system. Here the upper left block is the Jacobian part of TS equations with respect to the TS variables, $J_{TS,TS}$. The lower right block is the Jacobian part for the coupled-in-time EMT equations, $J_{EMT,EMT}$. The other two blocks are the jacobian blocks for the coupling between TS network voltages and the EMT boundary currents, $J_{TS,EMT}$ (upper right) and $J_{EMT,TS}$ (lower left).

Writing the EMT equations first followed by the TS equations, the linear system to solve at each Newton iteration is

$$\begin{bmatrix} J_{EMT,EMT} & J_{EMT,TS} \\ J_{TS,EMT} & J_{TS,TS} \end{bmatrix} \begin{bmatrix} \Delta X_{EMT} \\ \Delta X_{TS} \end{bmatrix} = \begin{bmatrix} -F_{EMT} \\ -F_{TS} \end{bmatrix}$$

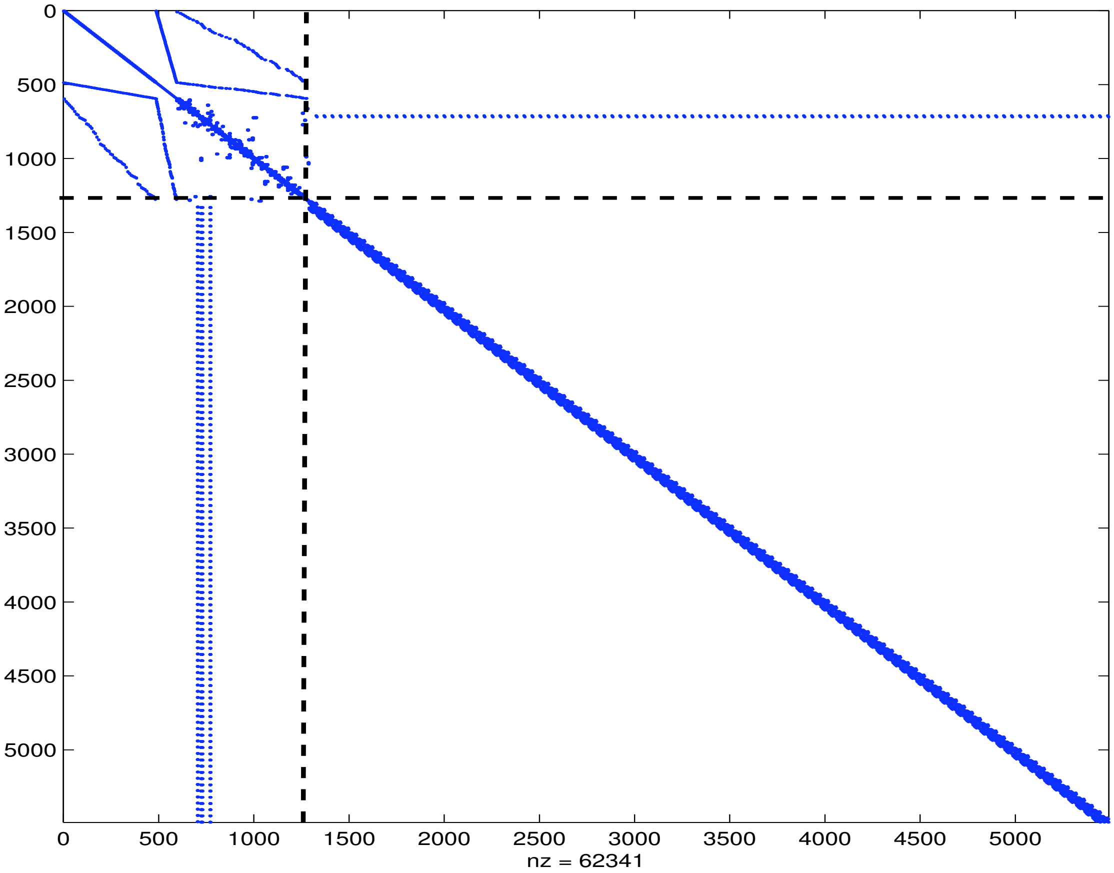


Figure 7.38. TSEMT Jacobian structure for 118 bus system

We experimented with three multiphysics preconditioners

1. Block-jacobi or *additive*

$$\begin{bmatrix} J^{-1}_{EMT,EMT} & \\ & J^{-1}_{TS,TS} \end{bmatrix}$$

2. Block Gauss-Siedel or *multiplicative*

$$\begin{bmatrix} J_{EMT,EMT} & \\ J_{TS,EMT} & J_{TS,TS} \end{bmatrix}^{-1}$$

which is

$$\begin{bmatrix} I & \\ & J_{TS,TS}^{-1} \end{bmatrix} \begin{bmatrix} I & \\ -J_{EMT,TS} & I \end{bmatrix} \begin{bmatrix} J_{EMT,EMT}^{-1} & \\ & I \end{bmatrix}$$

and

3. Schur complement based

$$\begin{bmatrix} I & -J_{EMT,EMT}^{-1}J_{EMT,TS} \\ & I \end{bmatrix} \begin{bmatrix} J_{EMT,EMT}^{-1} & \\ & S^{-1} \end{bmatrix} \begin{bmatrix} I & \\ J_{TS,EMT}J_{EMT,EMT}^{-1} & I \end{bmatrix}$$

where $S = J_{TS,TS} - J_{TS,EMT}J_{EMT,EMT}^{-1}J_{EMT,TS}$

The multiphysics preconditioner allow to user to choose individual linear solver, factorization methods and reordering strategies for TS and EMT. Looking at the block $J_{EMT,EMT}$ in 7.38, a *Reverse Cuthill Mckee* ordering would produce the least number of fill-ins for $J_{EMT,EMT}^{-1}$ since this block is already in a block-subdiagonal form. For TS3ph, based on the results from Chapter 5, ILU(6) with *quotient minimum degree* ordering could be used for $J_{TS,TS}^{-1}$. Thus the factorization of the TS and EMT blocks could be done efficiently.

The multiphysics preconditioners were found to be the best for TSEMT as seen from the results in tables 7.4 - 7.6.

For TS3ph, the linear solution strategy used was GMRES + ILU(6) preconditioner + *qmd ordering.*

As an example the following linear solver options were used for the best strategy (*fieldsplit multiplicative + lag*) in table 7.5

For TS3ph:

GMRES + ILU(6) preconditioner + compute preconditioner only once.

For TSEMT:

GMRES + fieldsplit preconditioner with type multiplicative

For TS part in TSEMT:

ILU(6) factorization + compute factorization once only at fault-on/fault-off times.

For the EMT part in TSEMT:

LU factorization with *rcm* ordering and update factorization only at fault-on/fault-off times.

Table 7.4. TS3ph-TSEMT timings results for the 9 bus system with various preconditioning strategies

Preconditioner Type	pre-TS3ph	TSEMT	post-TS3ph	Total
LU	3.31E-02	7.57E-01	1.06E-01	1.18E+00
LU + lag	3.88E-03	2.41E+00	1.04E-01	6.65E-01
ILU + 6	4.07E-03	8.21E-01	1.04E-01	1.17E+00
Fieldsplit additive	3.91E-03	4.79E-01	1.03E-01	7.80E-01
Fieldsplit additive + lag	4.23E-03	4.46E-01	1.08E-01	9.89E-01
Fieldsplit additive + ILU(6) for TS3ph + lag	3.82E-03	4.82E-01	1.04E-01	7.43E-01
Fieldsplit schur	4.91E-03	5.28E-01	1.04E-01	9.50E-01
Fieldsplit schur + lag	4.07E-03	1.03E+00	1.04E-01	1.33E+00
Fieldsplit schur + ILU(6) for TS3ph + lag	3.66E-03	5.30E-01	1.04E-01	7.11E-01
Fieldsplit multiplicative	3.79E-03	3.43E-01	1.03E-01	5.24E-01
Fieldsplit multiplicative + lag	**3.67E-03**	**3.10E-01**	**1.04E-01**	**4.51E-01**
Fieldsplit multiplicative + ILU(6) for TS3ph + lag	3.58E-03	3.45E-01	1.03E-01	5.25E-01

Table 7.5. TS3ph-TSEMT timings results for 118 bus system with various preconditioning strategies

Preconditioner Type	pre-TS3ph	TSEMT	post-TS3ph	Total
LU	1.35E-01	5.08E+00	3.47E-01	5.79E+00
LU + lag	1.40E-02	1.12E+00	4.25E-01	1.76E+00
ILU + 6	1.30E-01	5.45E+00	3.43E-01	6.15E+00
Fieldsplit additive	1.39E-02	3.95E-01	3.45E-01	9.47E-01
Fieldsplit additive + lag	1.31E-01	2.99E-01	3.84E-01	1.00E+00
Fieldsplit additive + ILU(6) for TS + lag	9.65E-03	4.07E-01	3.78E-01	9.33E-01
Fieldsplit schur	1.38E-02	4.48E-01	3.42E-01	9.91E-01
Fieldsplit schur + lag	1.39E-02	5.57E-01	3.03E-01	9.74E-01
Fieldsplit schur + ILU(6) for TS3ph + lag	9.75E-03	4.61E-01	2.98E-01	9.50E-01
Fieldsplit multiplicative	1.31E-01	2.86E-01	3.43E-01	9.87E-01
Fieldsplit multiplicative + lag	**9.51E-03**	**2.20E-01**	**3.37E-01**	**6.24E-01**
Fieldsplit multiplicative + ILU(6) for TS + lag	9.49E-03	2.89E-01	2.97E-01	6.53E-01

Table 7.6. TS3ph-TSEMT timings results for 1180 bus system with various preconditioning strategies

Preconditioner Type	pre-TS3ph	TSEMT	post-TS3ph	Total
LU	1.02E-01	6.33E+00	7.52E+00	1.44E+01
LU + lag	1.04E-01	1.84E+00	7.03E+00	9.41E+00
ILU + 6	2.24E-01	1.06E+01	6.92E+00	1.81E+01
Fieldsplit additive	1.02E-01	6.52E+00	7.17E+00	1.44E+01
Fieldsplit additive + lag	2.19E-01	1.53E+00	7.06E+00	9.25E+00
Fieldsplit additive + ILU(6) for TS3ph + lag	9.76E-02	3.70E+00	7.17E+00	1.15E+01
Fieldsplit schur	9.36E-02	5.34E+00	6.57E+00	1.26E+01
Fieldsplit schur + lag	1.06E-01	2.33E+00	7.05E+00	9.91E+00
Fieldsplit schur + ILU(6) for TS3ph + lag	9.74E-02	3.45E+00	7.04E+00	1.10E+01
Fieldsplit multiplicative	9.85E-02	5.21E+00	7.00E+00	1.29E+01
Fieldsplit multiplicative + lag	**9.71E-02**	**1.45E+00**	**7.22E+00**	**9.21E+00**
Fieldsplit multiplicative + ILU(6) for TS3ph + lag	9.96E-02	2.00E+00	7.36E+00	9.91E+00

7.6.3 Comparison of TS, TSEMT, and EMT run times.

Table 7.7 shows the comparison of the cpu run times for TS, TSEMT, and EMT for a 3 second three phase fault simulation resulting in a stable system. The computational burden for EMT grows with the system size and for the 118 bus system EMT takes about 30 seconds. TS3ph is the fastest for both cases while the TS3ph-

TSEMT simulator does not lag behind by much.

Table 7.7. Comparison of TS, TSEMT, and EMT run times

System size	Simulated time (sec)	TS3ph	EMT	Only TSEMT	TS3ph-TSEMT
9 bus	3	0.13	4.96	5.46	0.41
118 bus	3	0.36	30.1	4.87	0.53

CHAPTER 8

PARALLEL IMPLEMENTATION OF TS3PH, TSEMT, AND TS3PH-TSEMT

8.1 Introduction

Transient stability simulation for large-scale power systems is a computationally onerous task due to the need of the solution of large number of equations at each time step. A natural way to speed up is by this computation is to use parallel computing techniques, i.e., share the work load amongst multiple processors. Three algorithms for the parallel transient stability simulation have been proposed so far : parallel-in-space, parallel-in-time, and parallel-in space and time.

The parallel-in-space algorithms partition the given network into loosely coupled or independent subnetworks. Each processor is then assigned equations for a subnetwork. The partitioning strategy for the network division is critical for parallel-in-space algorithms to minimize the coupling between the subnetworks, i.e., to reduce the inter-processor communication, and balance the work load. Once a partitioning strategy is selected the next key thing is the solution of the linear system in each Newton iteration. Several linear solution schemes have proposed in the literature, of which the prominent are the *Very Dishonest Newton Method* and *Conjugate gradient*. Reference [13] uses the very dishonest Newton method in which the factorization of the jacobian matrix is done only when certain fixed number of iterations are exceeded. In [20], a decomposition of the network equations in a Block Bordered Diagonal Form (BBDF) is done first and then a hybrid solution scheme using LU and Conjugate gradient is used. Reference [19] solves the network by a block-parallel version of the preconditioned conjugate gradient method. The network matrix in [19] is in the Near Block Diagonal Form (NBDF).

Parallel-in-time approach was first introduced in [5]. The idea of this approach is to combine the differential and algebraic equations over several time steps, create a bigger system and solve them simultaneously using the Newton's method. All the equations for several integration steps are assigned to each processor.

Waveform relaxation methods [38], [17], [30] involve a hybrid scheme of space and time parallelization in which the network is partitioned in space into subsystems on each processor and several integration steps for each subsystem are solved independently to get a first approximation [23]. The results are then exchanged and the process is repeated. The advantage of this scheme is that each subsystem can use a different integration steps (multi-rate integration).

8.2 Cluster details

All the parallel runs were done on the cluster 'Karlin' hosted by Mathematics Department at Illinois Institute of Technology. This cluster has 16 nodes with 2 Intel Xeon 2.53 GHz processors per node. Each processor has 4 cores with 3 GB RAM for each node giving a total of 24 GB RAM for each node. Karlin uses 10 Gb/s Infiniband network.

8.3 Parallel TS3ph

For TS3ph, we adopt a parallel-in-space approach for the decomposition of differential and network equations. Such an approach was chosen to allow it to be used with TSEMT for TS3ph-TSEMT simulator. The decomposition of the equations is based on the partitioning of the network equations since the network forms the only coupling. The generator and the load equations are naturally decoupled as they are only incident at a bus.

8.3.1 Partitioning. The ParMetis package [41] was used for doing the network partitioning. The ParMetis package [41] is available as a plug-in with the PETSc library and can be downloaded and installed along with the installation of PETSc.

PETSc provides an interface for the ParMetis.

ParMetis requires an *adjacency matrix* whose elements comprise of 1s and 0s, where an element $A_{i,j}$ is 1 if vertex i is connected to vertex j. Along with the *adjacency matrix*, a weight can be also can be assigned to each vertex to account for different computational requirement. With vertex weights, ParMetis tries to minimize the edge cuts and also have the same amount of vertex weights in each sub-domain.

For TS3ph, the single-phase network connectivity matrix was used as the adjacency graph. Larger weights were assigned to the vertices having generators to account for the extra computation involved for the generator differential and algebraic equations.

When ParMetis solves this minimum edge-cut problem with weighted vertices, it returns a list of nodes assigned to each processor.

8.3.2 Preprocessing. Using the partitioning information, each processor then reads the ASCII files containing the generator, load data and initial conditions. Since the data in these files is for the entire network, each processor reads all the files but saves only the data needed for computing its own equations.

The three phase bus admittance matrix, Y_{bus}, for the complete network, which is stored in a binary file, is read only on one processor initially and then divided amongst processor. A processor gets the entire bus admittance matrix row if the bus belongs to its local subnetwork. Thus, the Y_{bus} matrix is stored row-wise in parallel. PETSc provides routines to read the matrix data stored in binary files on one or many processors and also provides routines for permuting the rows or column of the matrix in parallel. We used these routines for assembling parallel three phase Y_{bus} matrix.

Other preprocessing tasks include:

- Create and allocate parallel solution and residual vectors for the Newton method.
- Create generator, network and load subsystem solution and residual vectors (work vectors).
- Compute and allocate memory required for the Jacobian matrix.
- Set the linear part of the Jacobian matrix.

8.3.3 Equations and numerical solution. Using the parallel-in-space approach, each processor gets a subset of the generator, network and load equations for the complete network. The generator and the load equations are naturally decoupled since they are incident only on the local network bus and do not require communication with other processors. The network requires communication with other processors to compute its current mismatch equations.

The equations to be solved by each processor at each TS3ph time step are

$$F \equiv \begin{bmatrix} f^p(x^p, I^p_{dq}, V^p) \\ g^p(x^p, I^p_{dq}, V^p, V^{offproc}) \end{bmatrix} \tag{8.1}$$

where the superscript p denotes the processor number and $V^{offproc}$ denotes the network voltages required from other processors for computing g^p. The variables for each processor are

$$X \equiv \begin{bmatrix} x^p \\ I^p_{dq} \\ V^p \end{bmatrix}$$

At each time step, a Newton method is used for the solution of 8.1.

Table 8.1. Total number of variables in test systems for parallel TS3ph

Buses	Generators	Branches	Total Variables
1180	540	2085	13026
2360	1080	4670	26046
4720	2160	11340	52086

8.4 Parallel TS3ph performance results

The parallel TS3ph code was tested on three large systems. The details of these test systems are given in Appendix B. The TS3ph parallel performance tests were run for a 3 second simulation time with a fault applied at 0.1 seconds and removed at 0.2 seconds, similar to the sequential case. The time step used was 0.01667 seconds, or 1 cycle. Both direct (using parallel solver package MUMPS) and iterative methods were investigated and the iterative methods were found to be more scalable than the direct methods. As the iterative solver, the default PETSc iterative solver, GMRES, with different preconditioners was tested.

The two preconditioning schemes investigated to optimize the parallel performance are given next.

1. **Parallel Block-Jacobi + lag**: Since the Jacobian matrix for TS3ph is in the Nearly Bordered Block Diagonal Form (NBDF), the first preconditioner used was a parallel Block-Jacobi preconditioner. As an example, if the Jacobian matrix on two processors is

$$\begin{array}{c|cc|} [0] & J_1 & J_2 \\ \hline [1] & J_3 & J_4 \end{array}$$

then the parallel block-jacobi preconditioner is

$$\begin{array}{c} [0] \\ [1] \end{array}\left[\begin{array}{cc} J_1^{-1} & \\ \hline & J_4^{-1} \end{array}\right]$$

The factorization of J_1 and J_4 can then be done independently on each processor and no communication is required for building the preconditioner. Moreover, we also used a lagging preconditioner strategy that updates the numerical factorization only at disturbance on/off times. For a single processor, ILU(6) factorization was found as an optimal preconditioner since it has less memory access latency than a full LU factorization. Memory access latency here refers to the time required to access an element from the main memory. If the factored matrix on one processor is large (in terms of the number of non zeros) then there is a possibility that it cannot fit into the cache resulting in it being stored in the main memory. The access of elements from the main memory is much slower, in the order of hundreds of CPU cycles, which results in overall slowdown of the computation. For the multiprocessor case however this may not be true since the size of the diagonal block is smaller. Hence, we tested both ILU(6) factorization as well as full LU factorization on the blocks for the multiprocessor case.

2. **ASM+more number of blocks/processor**: The default parallel Block-Hacobi algorithm in PETSc uses one diagonal block per processor to form the preconditioner. PETSc also provides creating more number of diagonal block per processor via the parallel Block-Jacobi preconditioner. As an example, if

the Jacobian matrix on two processors is of the form

$$\begin{array}{c|cccc|}
 & J_{1a} & J_{1b} & & \\
 & & & J_2 & \\
[0] & J_{1c} & J_{1d} & & \\ \hline
[1] & & & J_{4a} & J_{4b} \\
 & J_3 & & & \\
 & & & J_{4c} & J_{4d}
\end{array}$$

, where the off-diagonal blocks J_{1b},J_{1c} , J_{4b}, and J_{4d} represents the Jacobin part for weak coupling within each subnetwork, then the preconditioner would be

$$\begin{array}{c|cccc|}
 & J_{1a}^{-1} & & & \\
[0] & & J_{1d}^{-1} & & \\ \hline
[1] & & & J_{4a}^{-1} & \\
 & & & & J_{4d}^{-1}
\end{array}$$

The division into more diagonal blocks on each processor is down row-wise in the parallel block-jacobi preconditioner. Such a row-wise division is suitable for structured grid problems. We tested such a row-wise division strategy but found it to be inefficient.

Hence, we investigated another strategy of dividing the diagonal block using ParMetis and found it to produce a better preconditioner when the size of the factored matrix is large enough not to fit into the cache. Such a scheme is embedded in the PETSc preconditioner *ASM* or *Additive Schwartz Preconditioner*. The *ASM* preconditioner in PETSc also allows having more number of diagonal blocks per processor with or without overlap between them. If the overlap is set to zero, the *ASM* preconditioner works as parallel block-jacobi. By default, *ASM* uses one block per processor. However, if more number of diagonal blocks/ per processor are requested then it passes the entire diagonal block to the partitioning package as the adjacency graph. We didn't extract this partitioning information but our conjecture is that the partitioning of the diagonal block further creates weakly connected subsystems which results in a good preconditioner. We tested this preconditioner for the largest test case, 4720 buses, and found to be better when the size of the factored matrix is large, i.e., with less cores.

A lagging preconditioning strategy was also used along with the *ASM* preconditioner.

Table 8.2. 1180 bus system TS3ph scalability results with GMRES + Block-Jacobi + ILU(6) + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	8.61	1.00
2	3	9.34	0.92
4	3	5.01	1.72
8	3	2.40	3.59
12	3	2.30	3.75
16	3	2.04	4.21
20	**3**	**1.70**	**5.05**
24	3	1.98	4.34

Note here that all these speed up results are compared to the best sequential strategy i.e. GMRES as the iterative linear solver with a Block-Jacobi preconditioner with ILU(6) factorization.

Table 8.3. 1180 bus system TS3ph scalability results with GMRES + Block-Jacobi + LU + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	9.42	0.91
2	3	7.35	1.17
4	3	3.89	2.22
8	3	1.99	4.32
12	3	2.29	3.76
16	3	2.04	4.21
20	**3**	**1.61**	**5.35**
24	3	1.98	4.34

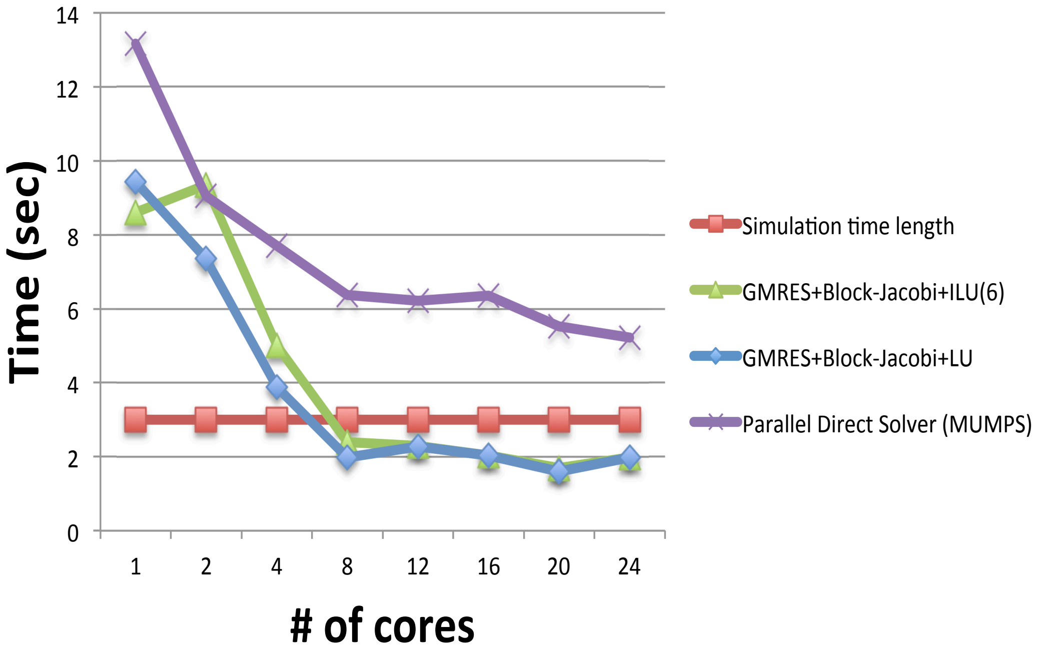


Figure 8.1. Comparison of TS3ph run-times for the 1180 bus system

Table 8.4. 1180 bus system TS3ph scalability results with parallel direct solver MUMPS with lagging numerical factorization

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	13.16	0.65
2	3	9.05	0.95
4	3	7.71	1.12
8	3	6.37	1.35
12	3	6.22	1.38
16	3	6.36	1.35
20	3	5.52	1.56
24	**3**	**5.22**	**1.65**

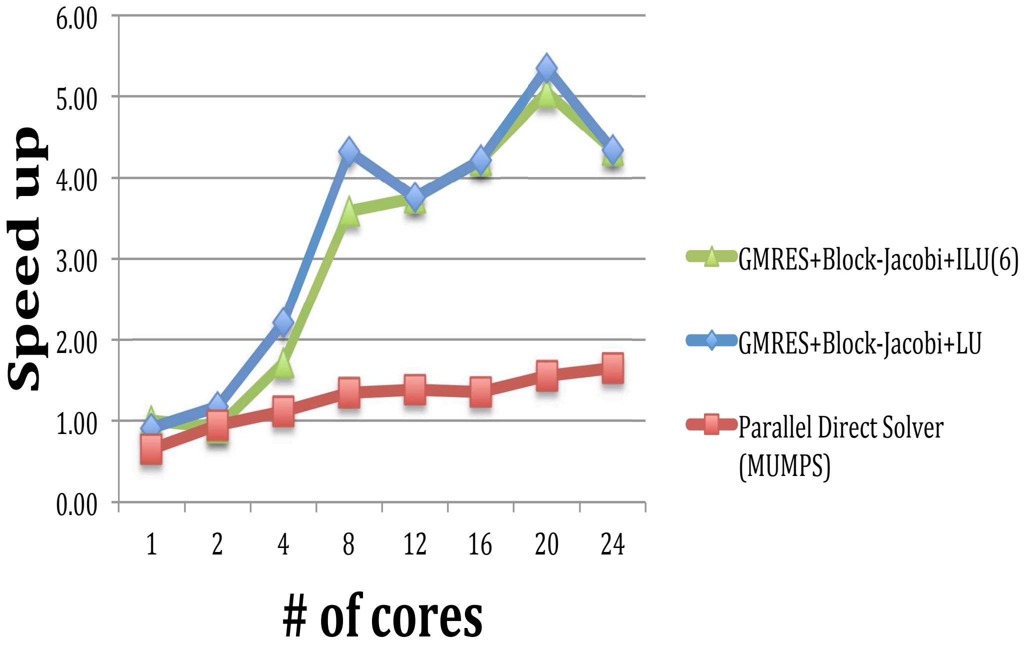


Figure 8.2. Comparison of TS3ph speedup for the 1180 bus system

Table 8.5. 2360 bus system TS3ph scalability results with GMRES + Block-Jacobi + ILU(6) + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	48.94	1.00
2	3	36.73	1.33
4	3	18.97	2.58
8	3	8.57	5.71
12	3	4.58	10.69
16	3	4.11	11.91
20	**3**	**3.61**	**13.55**
24	3	4.18	11.72
28	3	8.07	6.07
32	3	13.60	3.60

Table 8.6. 2360 bus system TS3ph scalability results with GMRES + Block-Jacobi + LU + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	67.78	0.72
2	3	30.10	1.63
4	3	14.61	3.35
8	3	6.12	8.00
12	3	4.25	11.52
16	3	3.51	13.93
20	**3**	**3.14**	**15.60**
24	3	4.13	11.84
28	3	7.95	6.16
32	3	13.62	3.59

Table 8.7. 2360 bus system TS3ph scalability results with parallel direct solver MUMPS with lagging numerical factorization

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	208.80	0.23
2	3	126.80	0.39
4	3	68.67	0.71
8	3	50.08	0.98
12	3	41.74	1.17
16	3	34.56	1.42
20	3	28.69	1.71
24	**3**	**25.92**	**1.89**
28	3	51.12	0.96

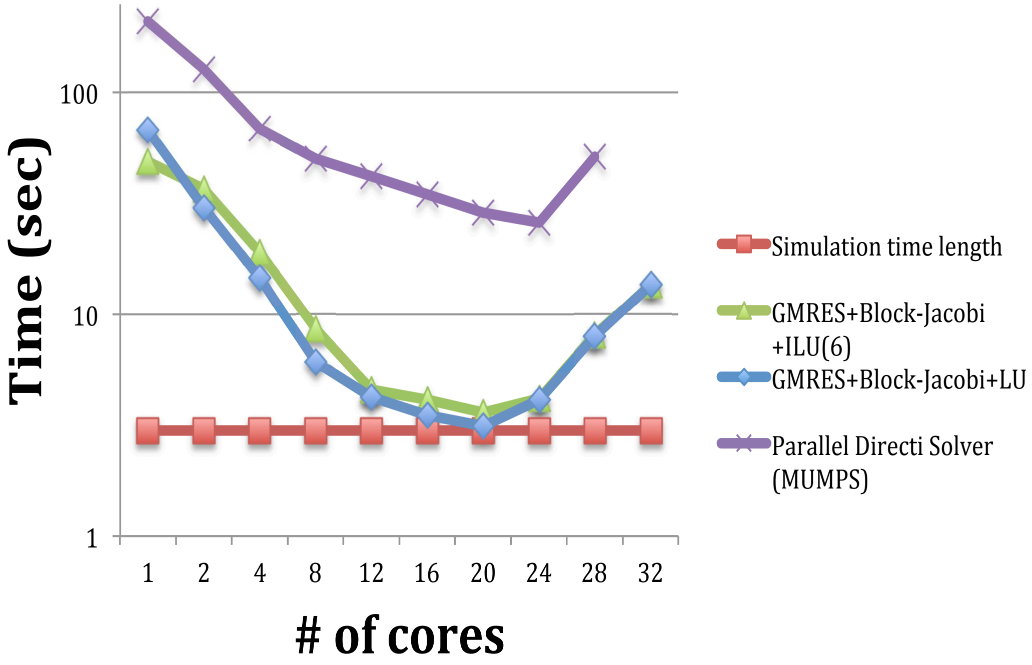


Figure 8.3. Comparison of TS3ph run-times for the 2360 bus system

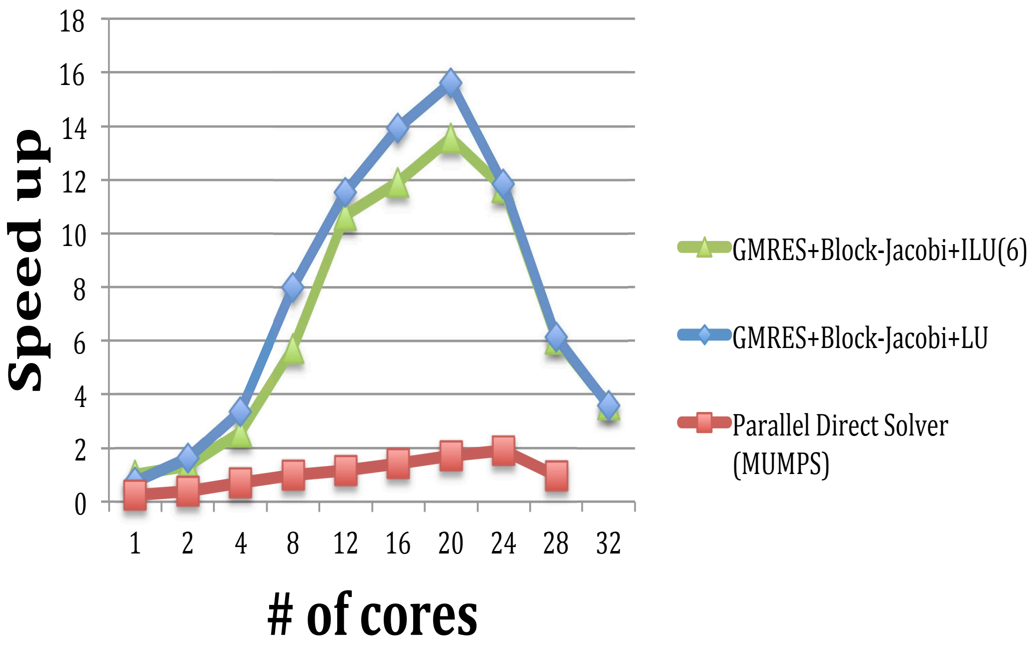


Figure 8.4. Comparison of TS3ph speedup for the 2360 bus system

Table 8.8. 4720 bus system TS3ph scalability results with GMRES + Block-Jacobi + ILU(6) + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	440.20	1.00
2	3	252.50	1.74
4	3	66.79	6.59
8	3	28.08	15.68
12	3	15.97	27.56
16	3	9.88	44.57
20	3	9.77	45.07
24	**3**	**7.15**	**61.61**
28	3	12.00	36.68
32	3	19.73	22.31
36	3	25.56	17.22
40	3	31.80	13.84

Table 8.9. 4720 bus system TS3ph scalability results with GMRES + Block-Jacobi + LU + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	606.80	0.73
2	3	159.90	2.75
4	3	49.45	8.90
8	3	20.08	21.92
12	3	12.66	34.77
16	3	8.73	50.44
20	3	6.83	64.46
24	**3**	**6.67**	**66.01**
28	3	11.83	37.21
32	3	18.20	24.19
36	3	25.57	17.22
40	3	29.99	14.68

Table 8.10. 4720 bus system TS3ph scalability results with GMRES + Block-Jacobi with 2 blocks/core + LU + lagging preconditioner

# of cores	Simulation time length	Execution time (sec)	Speed up
1	3	296.4	1.49
2	3	89.43	4.92
4	3	41.12	10.71
8	3	129.7	3.39

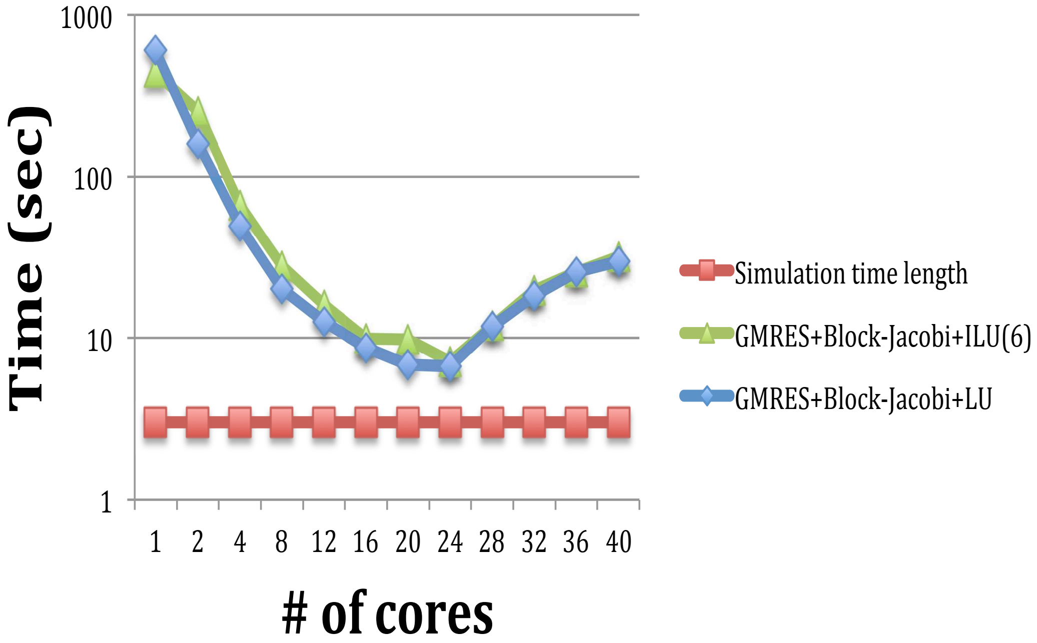


Figure 8.5. Comparison of TS3ph run-times for the 4720 bus system

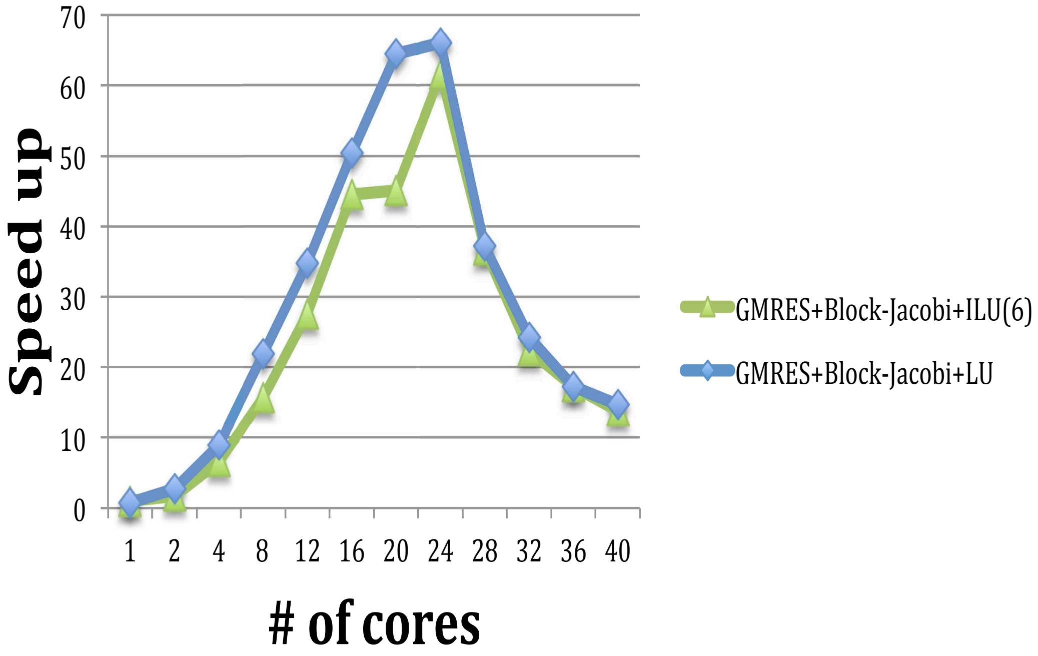


Figure 8.6. Comparison of TS3ph speedup for the 4720 bus system

8.4.1 Summary of parallel TS3ph results.

The results of the parallel performance of TS3ph can be summarized as follows

1. For all the test systems, the iterative solver GMRES with a Block-Jacobi preconditioner with LU factorization on the block result in the maximum speed up obtained.

2. For the 1180 bus system, the best time obtained is 1.61 seconds on 20 cores using Block-Jacobi preconditioner with LU factorization on the block. Note here that this is less than the simulation time length, i.e. 3 seconds. Thus, the TS3ph simulator runs faster than real-time for this system on 20 cores.

3. For all the test cases, as observed from results in Chapter 7, ILU(6) factorization is better than doing an LU factorization on single core. This can be explained from the number of nonzeros in the factored ILU(6) and LU matrices. Since LU produces more number of non zeros than ILU(6) hence more computation is needed. More elements in the factored elements also entails retrieving elements from the main memory which is more expensive. Hence, LU is more expensive than ILU(6) for large factored matrices. For the 1180 bus system, the difference in run-times using ILU(6) and LU is about 0.8 seconds while that for the 2360 bus system is about 19 seconds.

4. For all the test cases, the iterative solver GMRES with a Block-Jacobi preconditioner was found to be more efficient than the parallel direct solver MUMPS.

5. PETSc's LU and ILU data structures store the L matrix elements from top to bottom and the U elements from bottom to top [63]. This rearrangement of the data structure was motivated from the way elements are accessed during triangular solves. This revised LU data structure allows the elements of the factored matrix to be accessed in a stream which provides much better cache

performance. Hence, for single core runs, the execution time using LU or ILU(6) using PETSc's factorization routines is much faster than using MUMPS.

6. Although the test systems are doubled in terms of the number of buses, there is a seven-fold and seventy-fold increase in the execution times for the 2360 bus and 4720 bus systems respectively on a single core. These higher single core execution times can be explained by the number of non-zeros in the factored matrices. As seen in Table 8.11, although the non-zero elements in the Jacobian matrices roughly double, the non-zeros in the factored matrices increase by a factor of 6.27 and 45.83 respectively for the 2360 and 4720 bus systems respectively. In addition to the increased non-zeros in the factored matrices, matrix-vector multiplies during the GMRES iterations also contribute to the higher execution times.

Table 8.11. Comparison of non-zero elements for the three test systems

# of buses	J(nnz)	J^{-1}(nnz)	J(nnz)/J_{1180}(nnz)	J^{-1}(nnz)/J^{-1}_{1180}(nnz)
1180	227538	808609	1	1
2360	490986	5075201	2.15	6.27
4720	1126098	37066146	4.94	45.83

7. For the 2360 bus system, the best execution time obtained was 3.14 seconds using 20 cores. This execution time is also near to the real-time requirement of 3 seconds.

8. For the largest test case, 4720 buses, the maximum speed up obtained was about 66 with 24 cores. It should be noted here that this high superlinear speed is due to more time taken for execution on a single core. Due to the large matrix size, of the Jacobian and the factored matrices, the access of matrix elements from the main memory is slower and hence slows down the overall computation.

A better measure of speed up would be to compare the execution time with a 2-core or a 4-core run.

9. Block-Jacobi with 2 blocks/core produces a better preconditioner for the 4720 bus system test case. This is due to the size of the factored matrices, for each block, being smaller resulting in less fill-ins. For a single core, the execution time is about 1.5 times faster than doing an ILU(6) factorization on the entire matrix. Even on 4 cores, the speed up is about 10.71 while that from having 1 LU block is 8.9. We found that using more number of blocks for more cores resulted in a slow down.

10. The execution times for all the three test systems show a saturation above 24 cores. This can be attributed to more communication needed between the cores. We observed an increase in communication time required for the matrix-vector multiplies during the GMRES iterations as the number of cores increased. For the 2360 bus system, the time spent in communicating values was about 5%, 25%, and 40% on 8, 24, and 28 cores respectively. A faster network could possibly alleviate this saturation issue.

8.5 Parallel TSEMT

For parallel TSEMT equations, we propose a strategy of doing a spatial decomposition, i.e., parallel in space, for the external system equations and parallel-in-time, i.e., temporal decomposition for the coupled-in-time detailed system equations. The proposed parallel strategy was chosen from the following observations

- The external system is larger compared to the detailed system for large-scale power systems.
- The detailed system has coupled-in-time equations.
- The biggest part in the TS subsystem is the transmission network.
- Generators, Loads are incident at transmission nodes i.e. their equations are local.
- Communication between processors should be minimum and the work load for each processor should be balanced.

We also examined two more parallel strategies

- Do a space decomposition of external system equations and a space decomposition of detailed system equations
- Do a space decomposition of the external system equations and have all the coupled-in-time detailed system equations on one processor

After reviewing these strategies, the proposed strategy was chosen for parallelizing TSEMT because

- As coupled-in-time detailed system equations are solved, strategy 1 would require more communication.

- Strategy 2 could cause potential load unbalance.
- The detailed system being fairly small, only the detailed system variables for the last time step on this processor need to be communicated to the next processor via our proposed strategy.

Using our proposed parallel strategy, each processor needs to solve the external system equations for its subnetwork and a subset of the coupled-in-time detailed system equations. For example, if 100 EMT time steps need to be solved together and 2 processors are used then processor 0 is assigned the detailed system equations for the first 50 EMT time steps, while the equations for the next 50 EMT time steps are assigned to processor 1.

8.5.1 Partitioning. The ParMetis package was used for doing a space decomposition of the external system. As in TS3ph, the single phase connectivity matrix of the external system was used as the adjacency graph and larger vertex weights were used for buses having generators incident on them. The detailed system uses a parallel-in-time decomposition where the all the EMT equations within one TS time step are split among different processors. Thus, each processor gets the detailed system equations for a few EMT time steps.

8.6 Parallel TS3ph-TSEMT

A parallel version of the TS3ph-TSEMT simulator was also developed in this research work. The psuedo code for the parallel TS3ph-TSEMT simulator is as follows:

The setting up of the parallel TS3ph includes doing partitioning on the complete network using ParMetis, reading ASCII and binary data files and setting up different data structures, computing and allocating the solution, residual vectors and alloca-

Algorithm 1 Psuedo code for parallel TS3ph-TSEMT simulator

```
1. Set up parallel TS3ph
2. Run parallel TS3ph
3. If $t = t_{split}$ go to 4 else go to 2
4. Set up parallel TSEMT
5. Run parallel TSEMT
6. If merging conditions satisfied then go to 7 else go to 5
7. Transfer needed variables from parallel TSEMT to parallel TS3ph
8. Continue with parallel TS3ph till simulation end time
```

tion memory for the TS3ph Jacobian matrix. At pre-defined point in time, TS3ph is halted and TSEMT commences. The setting up of parallel TSEMT includes partitioning the external system using ParMetis, reading ascii and binary data files for the external and detailed system, TSEMT Jacobian allocation, etc. Currently, partitioning information from TS3ph is not reused for the external system in TSEMT. Reusing the TS3ph partitioning is a part of the future work. When the TSEMT simulator is terminated, all the necessary variables from the detailed and external system from different processors are passed to TS3ph.

8.7 Parallel TS3ph-TSEMT performance results

The TS3ph-TSEMT code was tested on two large systems: 1180 bus and 2360 bus system. Each of these systems has the detailed system as the radial system formed by buses 20-23 with three transmission lines and four loads. As a disturbance scenario, a three phase fault was applied on bus 21 at 0.1 seconds and removed at 0.2 seconds. TS3ph uses a time step of 1 cycle i.e. 0.01667 seconds and EMT uses a time step of 1/100th of a cycle i.e. 0.0001667 seconds. The total simulation time for these test cases is 3 seconds. TS3ph runs initially till 0.05 seconds after which TSEMT begins and runs until the merging algorithm based on boundary bus phasor voltage difference detects that TSEMT can be terminated. From the simulations it was found that for both the systems, TSEMT terminates at 0.233 seconds i.e. two cycles after the fault is cleared.

For all the test runs TS3ph uses GMRES with a parallel Block-Jacobi preconditioner with ILU(6) factorization and a quotient minimum degree ordering scheme for each diagonal block. GMRES was used for TSEMT and different preconditioners were experimented.

All the preconditioners experimented for the TSEMT part in TS3ph-TSEMT use a lagging preconditioner strategy, where the numerical factorization for building the preconditioner is done only at the beginning, fault on, and fault off times.

We experimented with three preconditioners for TSEMT:

1. **Direct parallel LU factorization using MUMPS:** MUMPS (Multifrontal Massively Parallel sparse direct solver) is a parallel LU factorization package that can be installed with PETSc.

2. **Block-Jacobi+ LU:** This preconditioning strategy uses the diagonal blocks of the Jacobian matrix on each processor as the preconditioner and then uses direct LU factorization on each block.

3. **Block-Jacobi + Fieldsplit:** Another preconditioning strategy that was experimented with was using multiphysics preconditioners, (additive and multiplicative fieldsplit as discussed in Chapter 7), on the Jacobian diagonal block. Since each diagonal block on the processor consists of a TS3ph and an EMT block further preconditioner customization can be done using additive or multiplicative preconditioners. For the TS3ph block an ILU(6) factorization with a quotient minimum degree ordering was used while for the EMT block a reverse cuthill mckee ordering with LU factorization was chosen.

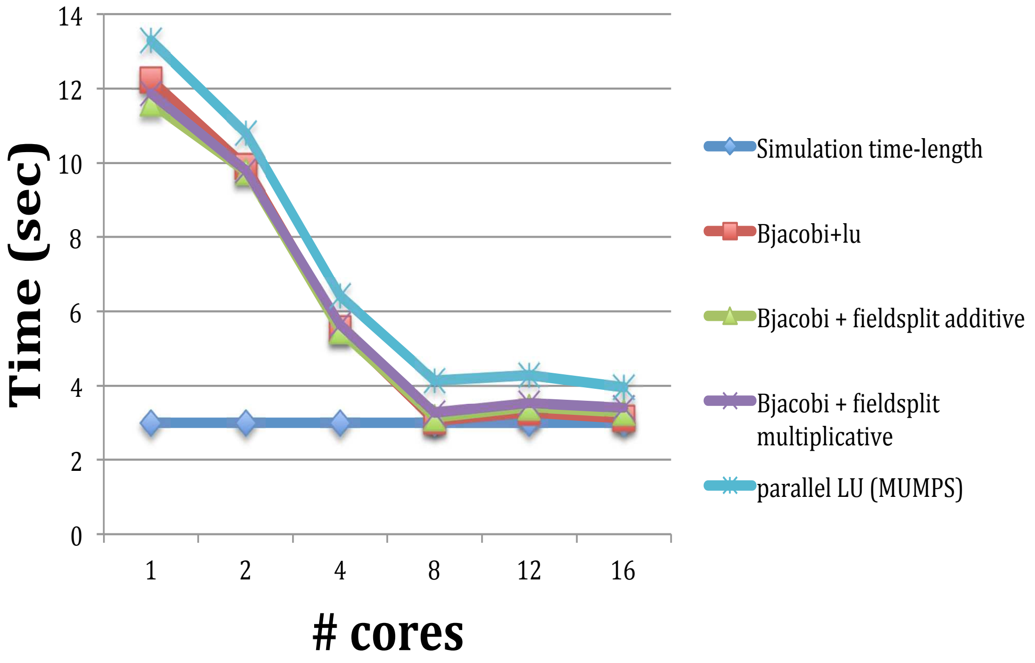


Figure 8.7. Comparison of TS3ph-TSEMT run-times for the 1180 bus system

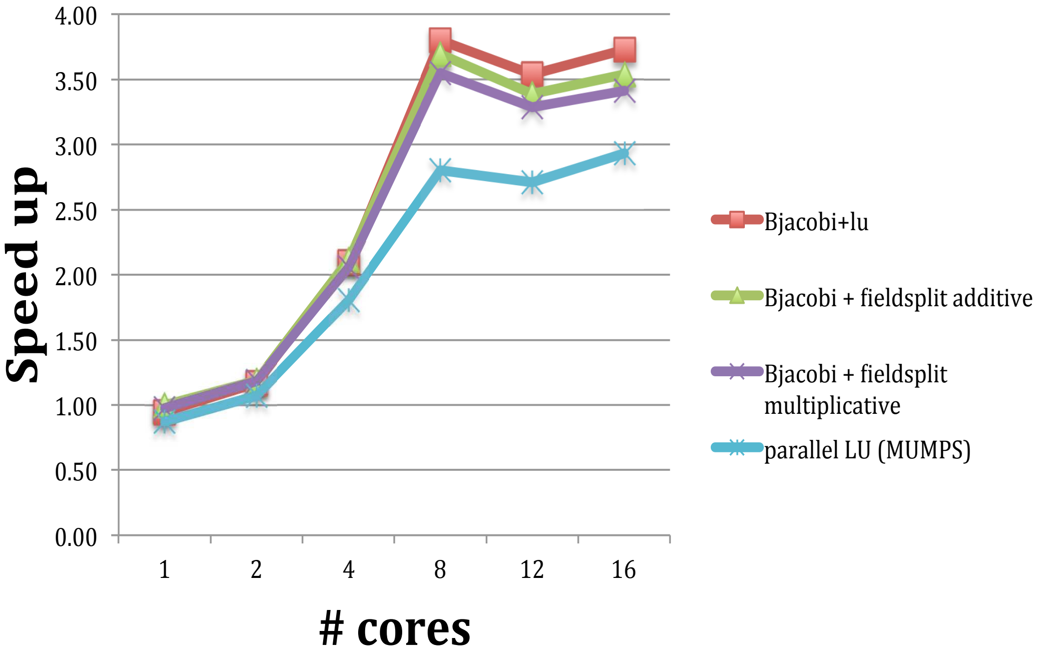


Figure 8.8. Comparison of TS3ph-TSEMT speedup for the 1180 bus system

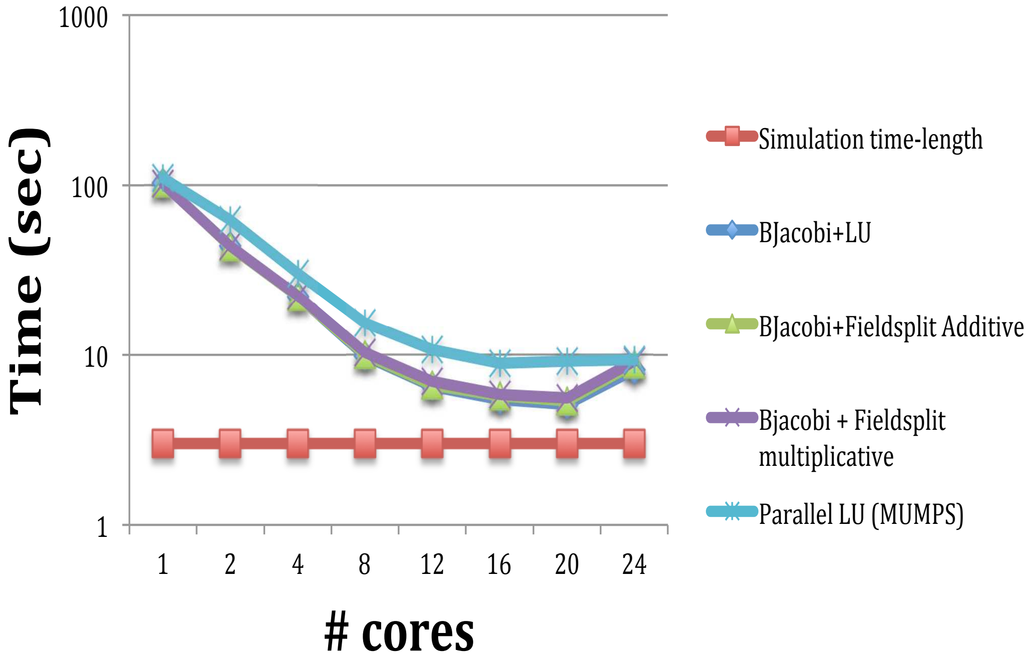


Figure 8.9. Comparison of TS3ph-TSEMT run-times for the 2360 bus system

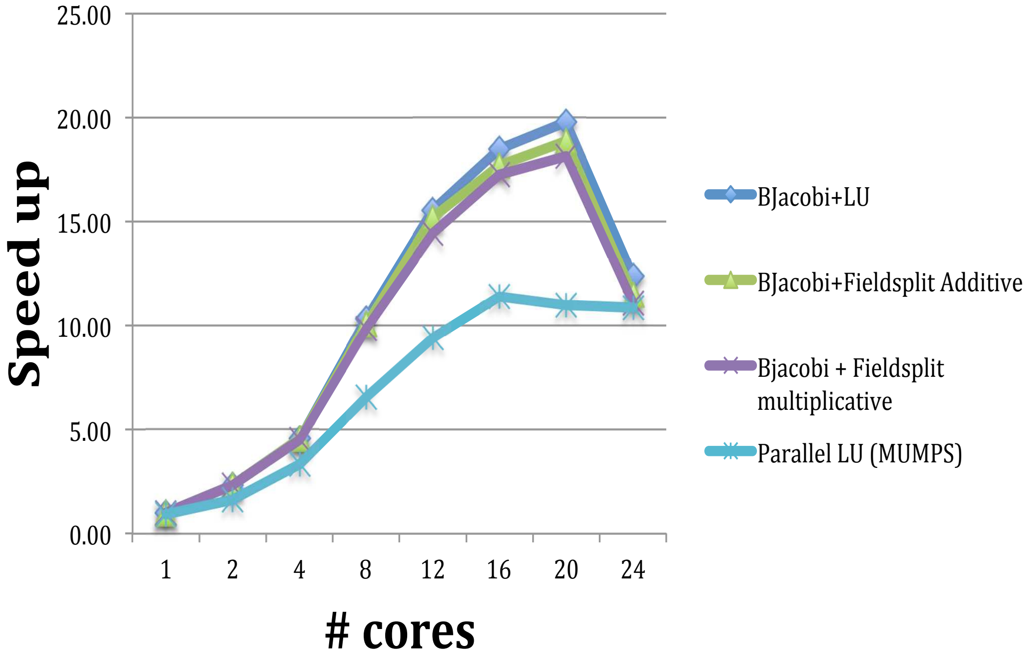


Figure 8.10. Comparison of TS3ph-TSEMT speedup for the 2360 bus system

8.7.1 Summary of TS3ph-TSEMT performance results. The performance of the TS3ph-TSEMT with different preconditioning strategies for the two test systems is shown in Figures 8.7-8.10. The results can be summarized as follows

1. For the 1180 bus system, the Block-Jacobi preconditioning results in the best performance with the lowest execution time of 3.11 seconds on 16 cores. The Block-Jacobi with LU and the fieldsplit preconditioners give identical speed up of approximately 3.5 on 16 cores. As compared to the Block-Jacobi preconditioning, the direct parallel solution using MUMPS has lower scalability of about 3.0 with 16 cores.

2. For the 2360 bus system, the performance of the Block-Jacobi preconditioner variants is the better compared to using a parallel direct solver. The Block-Jacobi preconditioners result in a speed up of close to 20 on 20 cores while that of the parallel direct solver is around 11.0.

3. Beyond 20 cores, a speedup saturation is observed for the 2360 bus system.

CHAPTER 9

CONCLUSIONS, CONTRIBUTIONS, APPLICATION AREAS, AND FUTURE WORK

9.1 Conclusions

Power system transient studies are done using transient stability simulators (TS) and electromagnetic transients simulators (EMT). TS simulators are used for studying slow electromechanical dynamics and use phasors for representation of voltages and currents. Use of phasors allows TS to run at a large time step, which makes TS attractive for a large-scale simulation. However, only fundamental frequency transients can be studied due to the fixed frequency assumption. On the other hand, an EMT can analyze transients over a wider frequency spectrum with detailed modeling of the equipments and using instantaneous voltage and current waveforms. The detailed modeling along with the time scale of interest force EMT simulators to use much smaller time step. Hence, EMT simulators are computationally inefficient for large-scale simulation.

A hybrid simulator (or combined electromechanical-electromagnetic transients simulator) combines the advantages of TS and EMT by modeling the large-scale system with phasor models and a smaller system with detailed systems. By using a large time step for TS and a smaller time step for EMT the computationally efficiency is not sacrificed while providing information about the slow dynamics in the external system and fast dynamics in the detailed system. The idea of a hybrid simulator provides an attractive approach for doing large-scale simulation of power systems with detailed models, such as relays or circuit breakers, in EMT.

The existing hybrid simulators use ‘explicit’ coupling to exchange data between TS and EMT at regular time intervals. At each data exchange, the data passed to EMT is either constant or based on extrapolated history data. While such an assumption may be sufficient for slow moving voltages and currents, it may not work for larger changes such as during a severe fault or voltage collapse scenario. Results for a serial interaction protocol were presented to justify that for large changes of voltages in the external system the serial interaction protocol may fail to converge.

This dissertation mainly focused on developing an implicitly coupled electromechanical and electromagnetic transients stability simulator.

9.2 Applications Areas of TS3ph-TSEMT simulator

The implicitly coupled TS3ph-TSEMT is a practical, robust, and computationally efficient option for simulation of large-scale power systems which require the detailed inspection of the fast dynamics of a critical area and a global view of the slow dynamics over a larger region. Several applications which require such a capability as discussed in the following subsections.

9.2.1 Identification of local voltage collapse and voltage collapse cascade.

As discussed in Chapter 1, section 1.4.1, voltage collapse is an important problem for utilities to deal with as the electric power systems operate near the limits in the deregulated environment. The identification of local voltage collapses and simulation of cascading events presents a challenge for the power system researchers. As already discussed in Chapter 1, local voltage collapse identification can assist in better measure of the transfer capability limits of the power system, while the simulation of the voltage collapse cascade phenomenon is important in planning preventive or corrective actions to avert a potential voltage collapse.

A phasor-based modeling approach, as done in the existing transient stability simulators and static analysis tools, may be incapable of capturing a voltage collapse or a voltage collapse cascade. This is due to the large frequency deviation caused during voltage collapse conditions rebutting the assumptions for phasor analysis. Reference [1] reports a case where the transient stability simulation prematurely terminates due to an impending voltage collapse. Thus, a transient stability simulator may be incapable of capturing the voltage collapse trajectories. On the other hand, an electromagnetic transients simulator has the capability to simulate voltage collapse trajectories [2]. However, an electromagnetic transients simulation is computationally inefficient and is not a practical approach. Instead, the implicitly coupled TSEMT simulator can provide a viable simulation method. By modeling the network experiencing voltage collapse conditions in the detailed system and the remainder of the network in the external system, the voltage collapse trajectories can be captured to determine whether the collapse is local or global. For a voltage collapse cascade simulation, several critical areas can be modeled within the detailed system to see the progression of a voltage collapse cascade as the events unfold.

9.2.2 Wide-area protection analysis. Currently, the testing of relays is done using EMT programs using simulated or recorded waveforms. Such a scheme uses local waveforms while the rest of the network is modeled as static or by reducing the network. As EMT programs are inefficient for large-scale power system analysis, a wide-area protection analysis cannot be done realistically. Such a wide-area analysis would be beneficial for the industry especially to study the role of relays in a voltage collapse cascade. By modeling the critical areas, where protection system needs to be studied, within the detailed system and the rest of the network in the external system an implicitly coupled TSEMT simulator can provide a computationally efficient simulation for wide-area protection analysis.

9.2.3 Modeling of power electronics for wind farms. The penetration of renewable energy sources, both at the transmission and distribution level, is increasing because of the promise of clean energy, regulatory incentives, and rapid advancement of renewable energy technology. Out of all the renewable energy sources, wind energy is the most promising due to its economic viability, and the growth in the installed wind generation over the last decade. As the penetration of the wind generated electricity in the grid increases, the flexible control of wind turbines and wind farms becomes more important.

Power electronics, such as converters, inverters, reactive compensation devices form an integral part of the wind farm control mechanism to match the characteristics of wind turbines with the requirements of grid connections, including frequency, voltage, control of active and reactive power, harmonics, etc [15]. Additional power electronics equipment is required for HVDC transmission lines that could be used for transporting electricity from off-shore wind farms. In an HVDC transmission, the low or medium AC voltage at the wind farm is converted into a high dc voltage on the transmission side and the dc power is transferred to the onshore system where the dc voltage is converted back into ac voltage. Traditionally, an electromagnetic transients program has been used to study the disturbances within the power electronic devices locally with the rest of the power system assumed to be in steady state. The implicitly-coupled TSEMT simulator could be used for studying such disturbances while allowing a dynamic representation of the rest of the power system. Such a study could aid in analyzing issues such as power quality issues which are not within the scope of a transient stability simulation.

9.3 Contributions

The contributions of this thesis work are as follows

9.3.1 A new implicitly coupled electromechanical and electromagnetic transients simulator. This research work proposed implicitly coupling the TS and EMT equations and solving them simultaneously in a larger set of equations. The TS and EMT equations are solved simultaneously rather than interfacing the solutions as found in the existing hybrid simulators. By solving the TS system equations and coupled-in-time EMT equations at each TS time step, the TS system equations can be discretized by a larger time step and EMT equations by a smaller one.

We also use a three phase fundamental frequency phasor model of the external system and not the positive sequence fundamental frequency phasor model which is used in the existing transient stability simulators.

9.3.2 A new parallel three phase transient stability simulator. A new parallel three phase transient stability simulator was developed in this research work which models all three phase of the network. The three phase model allows simulation of unbalanced faults such as single line-to-ground, phase-phase more realistically and study the effect on each phase.

9.3.3 A new strategy to terminate the combined electromechanical and electromagnetic transients simulation. A new strategy to terminate the combined electromechanical and electromagnetic transients simulation and continue with electromechanical transient stability simulation was proposed. This strategy is based on the fundamental frequency phasor voltage difference between the external and detailed system boundary buses. By monitoring these boundary bus voltages, the TSEMT simulation can be discontinued when the two are close to each other.

9.3.4 Investigating multiphysics preconditioners for speeding up TSEMT. Various multiphysics preconditioners such as block-jacobi, block-gauss-siedel were experimented with for speeding up TSEMT simulation and found to be more efficient than LU or incomplete LU with different factorization levels.

9.3.5 A parallel implementation of TSEMT. A novel approach for parallelizing the TSEMT equations was proposed. This approach uses a parallel-in-space decomposition of the external system equations for one TS time step and a parallel-in-time decomposition of the detailed system equations.

9.3.6 An Integrated combined electromechanical and electromagnetic transients simulator. An integrated electromechanical and electromagnetic transients simulator was developed using the high performance computing library PETSc in this research work. The simulator can work in four different modes

1. A parallel implicitly coupled electromechanical and electromagnetic transients simulator (TSEMT)
2. A parallel three phase transient stability simulator (TS3ph)
3. Parallel TS3ph-TSEMT
4. A sequential EMT simulator

9.4 Future Work

This research work built a framework for the combined electromechanical and electromagnetic transients simulator. A lot of development work still needs to be done to make this simulator a practical application

9.4.1 Network equivalents. For the combined TSEMT simulator to be faithful both reliable simulation and efficiency are important. The TSEMT simulator would be reliable if the fast dynamics of the detailed system and the relatively slow dynamics of the external system are accurate. During the first few cycles after the fault, the instantaneous voltages are qualitatively the same but there is a considerable difference in the boundary currents. For overcurrent relays, the first few cycles of the fault currents is the activation period. Hence the simulation of the currents needs to be accurate if relay models are included in the simulation. The network equivalent for the external system that is used in the work done is a thevenin equivalent of the external system derived at fundamental frequency. However such a simple fundamental frequency equivalent would be inappropriate if considerable waveform distortion and phase imbalance is present at the interface buses.

Research in the area of network equivalents has shown that frequency dependent network equivalents present a better picture of the external system to the EMT simulator [77], [35] ,[6] and the detailed system can be kept at minimum. The feature of frequency dependent network equivalents is also available in commercial electromagnetic transients simulator programs like EMTP and EMTDC. More recently fitting a rational function in s domain [26], [60] or z domain [75], [76] for computing the frequency dependent equivalent has been the dominant approach. These approaches need to be investigated to produce a better equivalent of the external system.

9.4.2 Additional equipment models for TS3ph. The basic generator,exciter, turbine governor and load models were implemented in this research work. More generator and especially exciter and turbine governor models need to be incorporated in TS3ph. Modeling of various three phase transformer configurations needs to be added.

9.4.3 Investigating the impact of different load models on individual phases. Experimentation with different load models on all three phases needs to be done to see how individual phase dynamics affect the overall system stability.

9.4.4 Additional equipment models for EMT. Different types of equipment models such as three phase and single phase induction motor loads, three phase transformers, various other generator, exciter models need to be incorporated. Most EMT simulators use distributed parameter or frequency dependent models of transmission lines and these need to be incorporated.

9.4.5 Modeling of circuit breakers in TSEMT. Modeling of circuit breaker operation in TSEMT needs some deliberation. Single-stepping EMT simulators use an adaptive time-stepping scheme for implementing breaker tripping operation where the time step is reduced until current zero is found. Since TSEMT solves multiple time-step EMT equations together, a strategy needs to be investigated to implement this breaker opening at current zero operation.

9.4.6 Power flow and initialization of dynamic variables. Currently the TSEMT simulator uses the power flow solution and initialization of dynamic variables from MatPower and script files written in MATLAB. These two processes need to be implemented in C code starting from reading the raw data file to initializing the dynamic variables.

9.4.7 Partitioning strategy for TSEMT. Currently the partitioning for TSEMT is based on the adjacency graph of the partial TS network i.e. the external system adjacency graph. Other strategies to obtain a better partitioning for TSEMT need to be investigated.

9.4.8 Investigating blocked factorization schemes. The network Y matrix for TS3ph is a 6 X 6 block matrix. Hence, a blocked factorization scheme can be used for it with a block size of 6. Based on the experience working with the PETSc library, blocked factorization methods provide better memory access for triangular solves and matrix-vector multiplies. Incorporating blocked factorization schemes for TS3ph and TSEMT should be investigated.

APPENDIX A

PORTABLE EXTENSIBLE TOOLKIT FOR SCIENTIFIC COMPUATION (PETSC)

The Portable, Extensible Toolkit for Scientific Computation (PETSc) is a suite of data structures and routines that provide the building blocks for the implementation of large-scale application codes on parallel (and serial) computers. PETSc uses the MPI standard for all message-passing communication. PETSc includes an expanding suite of parallel linear, nonlinear equation solvers and time integrators that may be used in application codes written in Fortran, C, C++, Python, and MATLAB (sequential). PETSc provides many of the mechanisms needed within parallel application codes, such as parallel matrix and vector assembly routines. The library is organized hierarchically, enabling users to employ the level of abstraction that is most appropriate for a particular problem. By using techniques of object-oriented programming, PETSc provides enormous flexibility for users.

PETSc consists of a variety of libraries (similar to classes in C++). Each library manipulates a particular family of objects (for instance vectors) and the operations one would like to perform on the objects. The objects and operations in PETSc are derived from our long experiences with scientific computation.

Each object consists of an abstract interface (simply a set of calling sequences) and one or more implementations using particular data structures. Thus, PETSc provides clean and effective codes for the various phases of solving applications, with a uniform approach for each class of problems. This design enables easy comparison and use of different algorithms (for example, to experiment with different Krylov subspace methods, preconditioners, or truncated Newton methods). Hence, PETSc provides a rich environment for modeling scientific applications as well as for rapid algorithm design and prototyping. The libraries enable easy customization and extension of both algorithms and implementations. This approach promotes code reuse and flexibility, and separates the issues of parallelism from the choice of algorithms. The PETSc infrastructure creates a foundation for building large-scale applications.

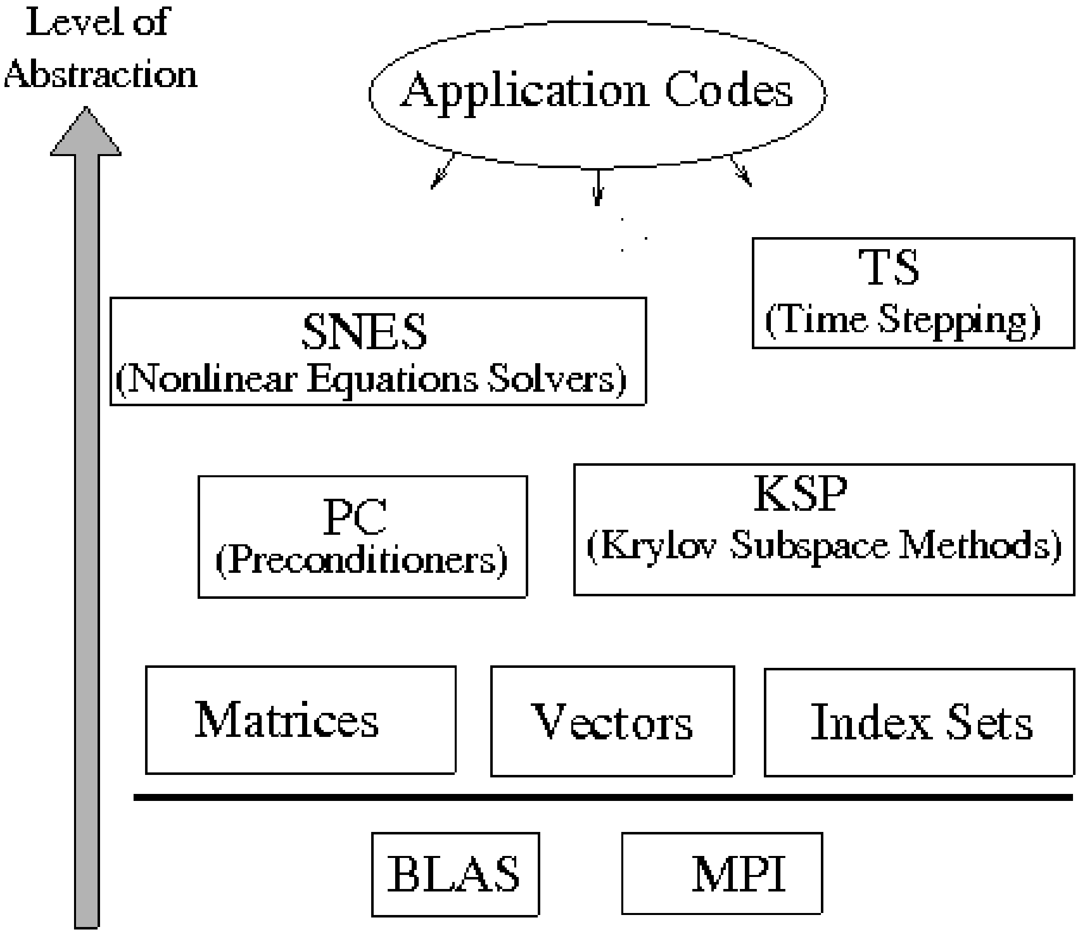


Figure A.1. Organization of the PETSc library [8]

A.1 PETSc features

PETSc is an open source package for numerical solution of large-scale applications and is free for anyone to use (BSD-style license). It runs on operating systems such as Linux, Microsoft Windows, Apple Macintosh, Unix operating systems. It can be used from within the Microsoft Developers Studio. PETSc can be configured to work with real or complex data types (not mixed though), single or double precision, and 32 or 64 bit integers. It has been tested on a variety of tightly coupled parallel architectures such as Cray XT/5, Blue Gene/P, Earth Simulator, and also on loosely coupled architectures such as networks of workstations.

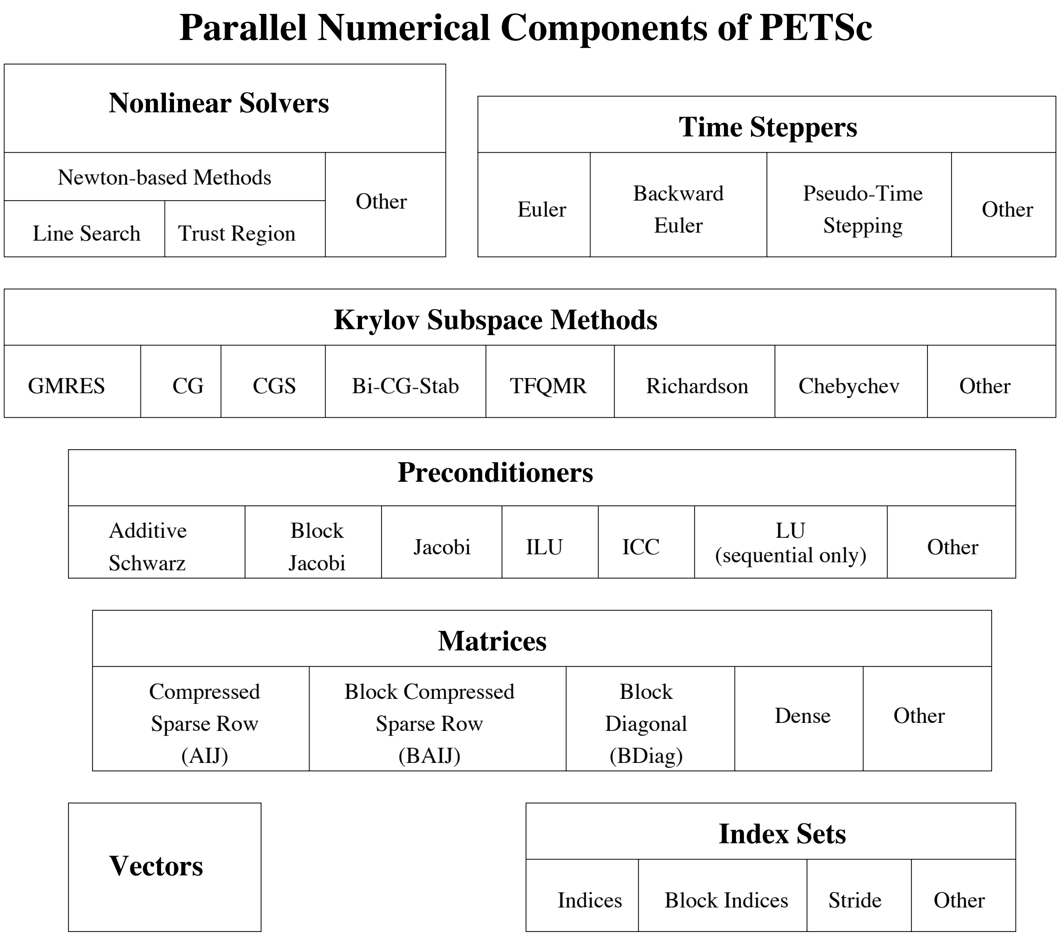


Figure A.2. Numerical Libraries of PETSc [8]

PETSc uses a plug-in philosophy to interface with external softwares. Various external softwares such as SuperLU, SuperLU_Dist, ParMetis, MUMPS, PLAPACK, Chaco, Hypre, etc., can be installed with PETSc. PETSc provides an interface for these external softwares so that they can be used in PETSc application codes.

Allowing the user to modify parameters and options easily at runtime is very desirable for many applications. For example, the user can change the linear solution scheme from GMRES to direct LU factorization, or change the matrix storage type, or preconditioners, via run time options. If an application uses a large number of parameters then these can be also supplied by via a text file which is read when the

PETSc code begins.

Debugging is one of the most pain-staking task in application code development. PETSc provides various features to ease the debugging process. Various debuggers such as gdb, dbx, xxgdb, etc., can be used for debugging PETSc application codes. The debugger can be either activated at the start of the program or when an error is encountered. Morever, a subset of processes can be also selected for debugging parallel application codes. In addition, the widely used package Valgrind can be used for detecting memory errors. Jacobian computation for the solution of nonlinear system via Newton's method is cumbersome and a great deal of time and effort can be spent in debugging the Jacobian. PETSc provides run time options to check the user Jacobian entries by comparing it with a finite difference approximated Jacobian.

PETSc automatically logs object creation, times, and floating-point counts for the library routines. Users can easily supplement this information by monitoring their application codes as well. The users can either log their routines, called an *event* logging, or multiple sections of the code, called *stage* logging.

A.2 PETSc use in the current research work

All the simulators developed in this research work were built using the PETSc library. A description of the libraries and the components used from PETSc is detailed below

A.2.1 Use of Vec and Mat library. The vector (Vec) and matrix (Mat) libraries were used in the simulators to store the solution vectors, right hand sides of the nonlinear functions as well as the various matrices needed such as the Jacobian, linear part of the generator differential equations, adjacency graph.

A.2.2 Network partitioning. The network partitioning for TS and TSEMT was done using ParMetis package. PETSc provides an interface for ParMetis so that the users can use PETSc data structures and routines to access ParMetis functions. For the TS and TSEMT network partitioning, the per-phase network graph, which is the connectivity graph, was provided to ParMetis.

A.2.3 SNES library. The Scalable Nonlinear Equation Solver (SNES) library was used for developing TS, EMT, and TSEMT simulators. Eventhough, these applications involve differential equations, a manual discretization (using implicit-trapezoidal scheme) was done and the resultant nonlinear functions solved using SNES. SNES requires two callback routines to evaluate the nonlinear function $f(x)$ and the Jacobian $J(x)$. The linear solver and the preconditioner can be set at run-time and we experimented with various native as well as third-party linear solvers and pre conditioners available with PETSc.

APPENDIX B

TEST SYSTEMS

B.1 WECC 9-bus system data

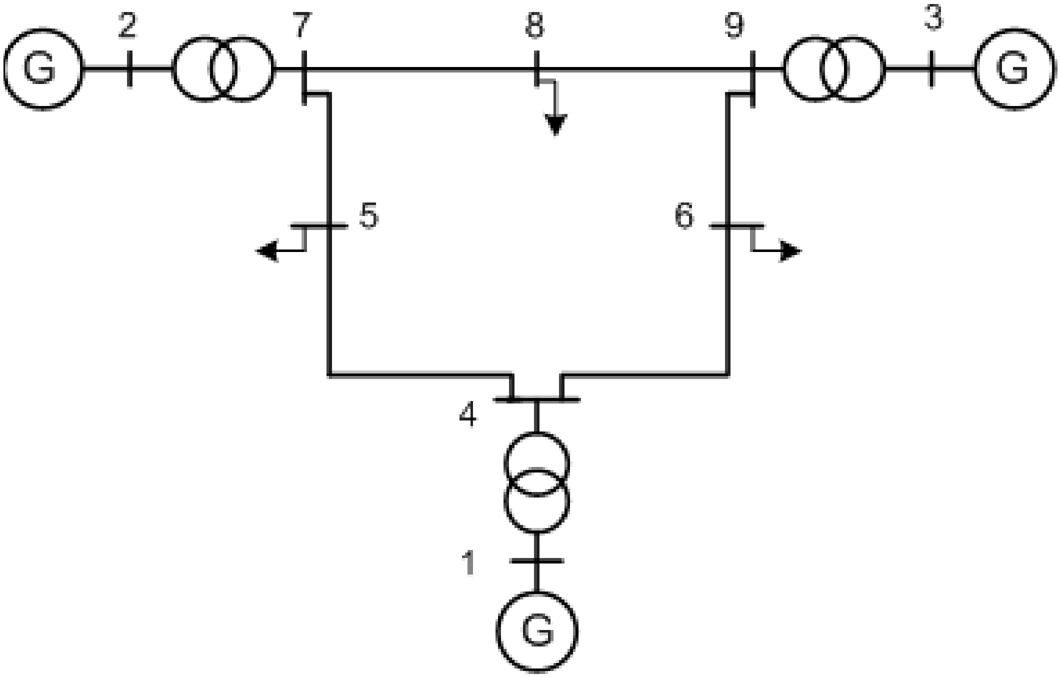


Figure B.1. WECC 9-bus system

Table B.1. 9-bus system generation and load data

Bus	P_{gen}(MW)	Q_{gen}(MVAr)	P_{load} (MW)	Q_{load} (MVAr)
1	71.6	27	0	0
2	163	6.7	0	0
3	85	-10.9	0	0
4	0	0	0	0
5	0	0	125	50
6	0	0	90	30
7	0	0	0	0
8	0	0	100	35
9	0	0	0	0

B.2 TS3ph larger test systems Three large power systems were created for testing TS3ph by duplicating the 118 bus system. To ensure that the individual 118 bus areas are connected, we used 5 randomly chosen tie lines between each area. Thus

Table B.2. 9-bus system branch data

From	To	R (pu)	X (pu)	B (pu)
1	4	0.0001	0.0576	0.0001
2	7	0.0001	0.0625	0.0001
3	9	0.0001	0.0586	0.0001
4	5	0.01	0.085	0.176
4	6	0.017	0.092	0.158
5	7	0.032	0.161	0.306
6	9	0.039	0.17	0.358
7	8	0.0085	0.072	0.149
8	9	0.0119	0.1008	0.209

Table B.3. 9-bus system machine data

Parameter	Bus 1	Bus 2	Bus 3
$T_{d0}^{'}$	8.96	8.5	3.27
$T_{q0}^{'}$	0.31	1.24	0.31
$T_{d0}^{''}$	0.05	0.037	0.032
$T_{q0}^{''}$	0.05	0.074	0.079
H	22.64	6.47	5.047
D	0	0	0
X_d	0.146	1.75	2.201
X_q	0.0969	1.72	2.112
$X_d^{'}$	0.0608	0.427	0.556
$X_q^{'}$	0.0608	0.65	0.773
$X_d^{''} = X_q^{''}$	0.05	0.275	0.327

each 118 bus system area is connected to every other by 5 tie lines.

B.3 TSEMT larger test systems Two large power systems 1180 bus and 2360 bus system were created for testing TSEMT by duplicating the 118 bus system as done for TS3ph. Both these test systems have the detailed region consisting of the radial connection formed by buses 20.21,22,23. This radial connection has three

Table B.4. 9-bus system exciter data

Parameter	Bus 1,2,3
K_A	20
T_A	0.2
K_E	1.0
T_E	0.314
K_F	0.063
T_F	0.35

Table B.5. TS3ph large-case test system inventory

Scale	Buses	Generators	Branches
10x	1180	540	1860
20x	2360	1080	2720
40x	4720	2160	5440

transmission lines and load at each bus. There are no generators in the detailed system.

APPENDIX C

KRYLOV SUBSPACE AND GMRES

Krylov subspace iterative methods are the most popular among the class of iterative methods for solving large linear systems. These methods are based on projection onto subspaces called Krylov subspaces of the form $b, Ab, A^2b, A^3b, \ldots$. A general projection method for solving the linear system

$$Ax = b \tag{C.1}$$

is a method which seeks an approximate solution x_m from an affine subspace $x_0 + K_m$ of dimension m by impositng

$$b - Ax_m \perp L_m$$

where L_m is another subspace of dimension m. x_0 is an arbitrary initial guess to the soution. A krylov subspace method is a method for which the subspace K_m is the Krylov subspace

$$K_m(A, r_0) = \mathrm{span}\{r_0, Ar_0, A^2r_0, A^3r_0, \ldots, A^{m-1}r_0\}$$

where $r_0 = b - Ax_0$. The different versions of Krylov subspace methods arise from different choices of the subspace L_m and from the ways in which the system is preconditioned.

The Generalized Minimum Residual Method (GMRES) is a projection method based on taking $L_m = AK_m(A, r_0)$ in which K_m is the m-th krylov subspace. This technique minimizes the residual norm over all vectors $x \in x_0 + K_m$. In particular, GMRES creates a sequence x_m that minimizes the norm of the residual at step m over the mth krylov subspace as follows [25]

$$||b - Ax_m||_2 = min||b - Ax||_2 \tag{C.2}$$

At step m, an arnoldi process is applied for the mth krylov subspace to generate the next basis vector. When the norm of the new basis vector is sufficiently small, GMRES solves the minimization problem

$$y_m = argmin||\beta e_1 - \bar{H_m} y||_2$$

where $\bar{H_m}$ is the $(m+1)xm$ upper Hessenberg matrix.

The GMRES algorithm becomes impractical when m is large because of the growth of memory and computational requirements hence a restarted GMRES approach is used.

APPENDIX D

PRECONDITIONERS

A *preconditioner* is a matrix which transforms the linear system

$$Ax = b$$

into another system with a better spectral properties for the iterative solver. For GMRES, a clustered eigen structure (away from 0) often results in rapid convergence, particularly when the preconditioned matrix is close to normal. If M is the preconditioner matrix, then the transformed linear system is

$$M^{-1}Ax = M^{-1}b \tag{D.1}$$

Equation D.1 is refered to as being preconditioned from the left, but one can also precondition from the right

$$AM^{-1}y = b, \qquad x = M^{-1}y \tag{D.2}$$

or split preconditioning

$$M_1^{-1}AM_2^{-1}y = M_1^{-1}b, \qquad x = M^{-1}y \tag{D.3}$$

where the preconditioner is $M = M_1M2$.

When krylov subspace mathods are used, it is not necessary to form the preconditioned matrices $M^{-1}A$ or AM^{-1} explicitly since this would be too expensive. Instead, matrix-vector products with A and solutions of linear systems of the form $Mz = r$ are performed (or matrix-vector products with M^{-1} if this explicitly known).

Designing a good preconditioner depends on the choice of iterative method, problem characteristics, and so forth. In general a good preconditioner should be (a) cheap to construct and apply and (b) the preconditioned system should be easy to solve.

D.1 Sequential preconditioners

D.1.1 Exact LU preconditioner. The first preconditioner that was experimented with in this thesis work was the exact LU preconditioner where $M = LU(A)$. However, it was found that for larger systems this preconditioner is very expensive and consumes abut 90 % of the total time in the numerical factorization phase.

D.1.2 Level based Incomplete LU preconditioner. The extra fill-in introduced in the L and U matrices by the gaussian elimination is discarded in varying amount of degrees in the incomplete LU factorization method. The extra fill-ins can be discarded based on a threshold value or allowed level of fill-in. PETSc provides level based incomplete LU preconditioners and we tested these in this research work. A level of fill is attributed to each matrix entry that occurs in the incomplete factorization process. Fill-ins are dropped based on the value of level of fill. The initial level of fill of a matrix entry a_{ij} is defined as

$$lev_{ij} = \begin{cases} 0, & \text{if } a_{ij} \neq 0 \text{ or } i = j, \\ \infty, & \text{otherwise} \end{cases} \tag{D.4}$$

Each time an element is modified by the ILU process, its level of fill is updated according to

$$lev_{ij} = minlev_{ij}, lev_{ik} + lev_{kj} + 1$$

With ILU(l), all fill-ins whose level is greater than l are dropped where l is a nonnegative integer. Note that for $l = 0$, the no-fill ILU(0) preconditioner is obtained.

D.1.3 Multiphysics preconditioners. For TSEMT we experimented with multiphysics preconditioners which can be used for coupled systems having different physics of the form

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \tag{D.5}$$

where x and y represent the dynamics of different physics. The preconditioner for such a system can be constructed as either

1. *block-jacobi or additive*

$$\begin{bmatrix} A^{-1} & \\ & D^{-1} \end{bmatrix} \tag{D.6}$$

2. *Block-gauss-siedel or multiplicative*

$$\begin{bmatrix} A & \\ C & D \end{bmatrix}^{-1} \tag{D.7}$$

   which is

$$\begin{bmatrix} I & \\ & D^{-1} \end{bmatrix} \begin{bmatrix} I & \\ -C & I \end{bmatrix} \begin{bmatrix} A^{-1} & \\ & I \end{bmatrix}$$

3. Schur complement based

$$\begin{bmatrix} I & -A^{-1}B \\ & I \end{bmatrix} \begin{bmatrix} A^{-1} & \\ & S^{-1} \end{bmatrix} \begin{bmatrix} I & \\ CA^{-1} & I \end{bmatrix}$$

   where $S = D - CA^{-1}B$

Note here that the block-jacobi and block-gauss siedel preconditioners can be extended to systems having more than two physics while the schur-complement based preconditioner is only for a two physics system.

D.2 Parallel preconditioners

D.2.1 Parallel block-jacobi. With the jacobian matrix in a nearly bordered block diagonal form, the diagonal block on each processor can be used as a preconditioner.

For a two processor case, the parallel block-jacobi preconditioner becomes

$$\begin{array}{c} [0] \\ \\ [1] \end{array} \left[\begin{array}{cc} J_1^{-1} & \\ \hline & J_4^{-1} \end{array}\right]$$

D.2.2 Parallel block-jacobi with more number of blocks/processor. Another variation of the parallel block-jacobi preconditioner is to exploit the weak connectivity, if any, in the diagonal block and divide it further into strongly coupled sub-diagonal blocks. Such a preconditioner for a two processor case is given in D.2.2

$$\begin{array}{c} \\ [0] \\ [1] \\ \\ \end{array} \left[\begin{array}{cccc} J_{1a}^{-1} & & & \\ & J_{1d}^{-1} & & \\ \hline & & J_{4a}^{-1} & \\ & & & J_{4d}^{-1} \end{array}\right]$$

where the diagonal block is further divided into sub-blocks and only the sub-diagonal blocks are retained to construct the preconditioner.

APPENDIX E

TSEMT CODE ORGANIZATION

In this research work, an integrated tool for electromechanical and electromagnetic transients simualtion was developed. All the code is written in the C language using the development version of the PETSc library. This integrated tool can run in four different modes.

- A parallel implicitly couped electromechanical and electromagnetic transients simulator (TSEMT).
- A parallel three-phase electromechanical transient stability simulator (TS3ph).
- A parallel TS3ph-TSEMT.
- A sequential electromagnetic transient simulator (EMT).

While TSEMT, TS3ph, and TS3ph-TSEMT can run in parallel independently, the EMT implementation is done only in serial currently.

E.1 Code organization

The organization of the TSEMT code follows the subsystem division it has i.e a TS subsystem, an EMT subsystem, and a Boundary subsystem. The code for each of these subsystems is stored in their own sub-directory e.g. all the TS3ph related code is stored in the sub-directory TS-dir. This directory subdivision allows individual code to be compiled if the user wants to run only TS3ph or only EMT simulation. Furthermore, minor changes in the TS3ph and the EMT code needed to be made for TSEMT to reuse the code.

1. TSEMT main directory

   This is the top-level directory containing the code required for the TSEMT and TS3ph-TSEMT, makefiles for compiling TSEMT, TS, or EMT, and the top-level header file TSEMT.h. It also contains an options file which provides the

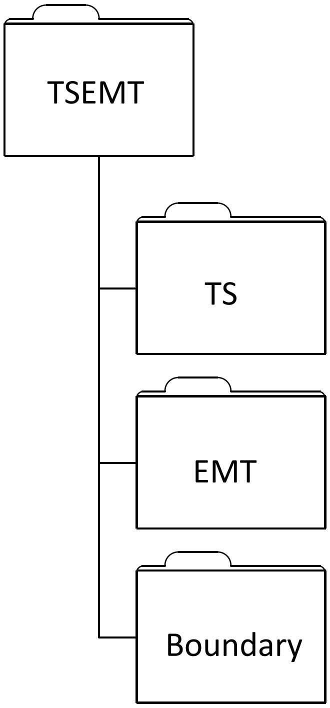


Figure E.1. TSEMT code organization

run-time options for setting PETSc linear and nonlinear solver options, location of data files needed for simulation, other parameters required.

2. Subdirectory TS-dir

   The directory contains the code for running TS3ph and includes its own top-level header file TS.h, and its own options file.

3. Subdirectory EMT-dir

   EMT-dir subdirectory includes the EMT code and has a structure and file names similar to TS3ph.

4. Subdirectory Bdry-dir

   The code for managing the TSEMT boundary equations and data is in the subdirectory Bdry-dir. It has its own header file which has data structures for the TS boundary portion, and the EMT boundary portion.

5. Subdirectory commonheaderfiles

   This sub-directory has header files for equipments which are common to both TS3ph and EMT such as the generator model GENROU or the exciter model IEEET1.

BIBLIOGRAPHY

[1] Abhyankar, S. G., and A. J. Flueck. “Simulating voltage collapse dynamics for power systems with constant power loads.”, *IEEE Power and Energy Society General Meeting.* (July 2008).

[2] Abhyankar S. G., and A. J. Flueck. “A new confirmation of voltage collapse via instantaneous time domain simulation.” *North American Power Symposium* (2009).

[3] Abhyankar, S. G., *Simulating voltage collapse dynamics for power systems with constant power load models.* Thesis. Illinois Institute of Technology, 2006.

[4] Abur, A., and H. Singh., “Time domain modeling of external systems for electromagnetic transients programs. *IEEE Transactions on Power Systems.* 8.2 (May 1993): 671-672.

[5] Alvarado, F. L., “Parallel solution of transient problems by trapezoidal integration.” *IEEE Transactions on Power Apparatus and Systems.* PAS-98 (May/June 1979): 1080-1090.

[6] Anderson, G. W., N. R.Watson, C. P. Arnold, and J. Arrillaga., “A new hybrid algorithm for analysis of HVDC and FACTS systems. *Proceedings of IEEE International Conference on Energy Management and Power Delivery*, 2, (November 1995): 462-467.

[7] *Art and Science of Protective Relaying.*
http://www.gedigitalenergy.com/multilin/notes/artsci/index.htm

[8] Balay, S., J. Brown, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, and B. F. Smith, and H. Zhang, *PETSc users manual.* ANL-95/11-3.1 (2010).

[9] Balay, S., J. Brown, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, and B. F. Smith, and H. Zhang, *PETSc users manual.* ANL-95/11-3.2 (May 2011).

[10] Balay, S., J. Brown, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, and B. F. Smith, and H. Zhang, *PETSc Web page.* (2011) http://www.mcs.anl.gov/petsc.

[11] Benzi., M., “Preconditioning techniques for large linear systems: A survey”. *Journal of Computational Physics* 182 (2002): 418 - 477.

[12] Chai., J. S., and A. Bose. “Bottlenecks in parallel algorithms for power system stability analysis”. *IEEE Transactions on Power Systems.* 8.1 (February 1993): 9 - 15.

[13] Chai., J. S., N. Zhu, A. Bose, and D.J. Tylavsky. “Parallel newton type methods for power system stability analysis using local and shared memory multiprocessors. *IEEE Transactions on Power Systems.* 6.4 (November 1991): 9 - 15.

[14] Chan., W., and L. A. Snider, “Electromagnetic electromechanical hybrid realtime digital simulator for the study and control of large power systems. *Proceedings of International Conference on Power System Technology*, 2 (December 2000): 783-788.

[15] Chen, Z., J. M. Guerrero, and F. Blaabjerg., "A review of the state of the art of power electronics for wind turbines". *IEEE Transactions on Power Electronics* 24.8 (August 2009): 1859-1875.

[16] Chiang, H. D., A. Flueck, K. Shah, and N. Balu. "CPFLOW: A practical tool for tracing system steady state stationary behavior due to load and generation variations. *IEEE Transactions on Power Systems.* 10 (May 1995): 623-634.

[17] Crow., M. L., and M. Ilic., "The parallel implementation of waveform relaxation methods for transient stability simulations". *IEEE Transactions on Power Systems.* 5.3 (August 1990): 922 - 932.

[18] Cutsem, T. V. "Voltage instability: phenomena, countermeasures and analysis methods. *Proceedings of the IEEE.* 88.2(Feb. 2000): 208-227.

[19] Decker, I.C., D. M. Falcao, and E. Kaszkurewicz. "Conjugate gradient methods for power system dynamic simulation on parallel computers". *IEEE Transactions on Power Systems.* 9.2 (May 1994): 629 - 636.

[20] Decker, I.C., D. M. Falcao, and E. Kaszkurewicz. " Parallel implementation of a power system dynamic simulation methodology using the conjugate gradient method". *IEEE Transactions on Power Systems.* 7.1 (February 1992): 458 - 465.

[21] Dobson, I., H. -D. Chiang, J. S. Thorp, and L. Fekih-Ahmed. "A model of voltage collapse in electric power systems. *Proceedings of IEEE Conference on Decision and Control.* (1998): 2104-2109.

[22] Dommel, H. W., *Electromagnetic transients program reference manual: EMTP theory book.* Portland, OR: Bonneville Power Administration, August 1986.

[23] Falcao., D., "High performance computing in power system applications".

[24] Fang, T., Y. Chengyan, W. Zhongxi, and Z. Xiaoxin., "Realization of electromechanical transient and electromagnetic transient real time hybrid simulation in power system. *Proceedings of IEEE Power Engineering Society Transmission and Distribution Exhibit.: Asia and Pacific*, (2005): 16.

[25] Flueck, A. J. *Advances in numerical analysis of nonlinear dynamical systems and the application to transfer capabiliy of power systems.* Diss. Cornell University, 1996.

[26] Gustavsen, B., and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting. *IEEE Transactions on Power Delivery.* 14.3 (July 1999): 1052-1061.

[27] Hansen, L. H., Helle, L., Blaabjerg, R. E., Munk-Nielsen, S., Bindner, H., Sorensen, P., and B. Bak-Jensen., *Conceptual survey of Generators and Power Electronics for Wind Turbines.* Riso National Laboratory, Roskilde, Denmark, (December 2001).

[28] Heffernan, M.D., K.S. Turner, J. Arrillaga, and C.P. Arnold., "Computation of AC-DC system disturbances:Part I,II,and III." *IEEE Transactions of Power Apparatus Systems.* 100.11 (November 1981) 4341-4363.

[29] Henville C., Folkers R., Heibert A., Weirckx R., "Dynamic simulation challenges protective performance
http://uyak03.files.wordpress.com/2008/02/sistem tenaga.pdf.

[30] Hou L., and A. Bose., "Implementation of the waveform relaxation algorithm on a shared memory computer for the transient stability problem". *IEEE Transactions on Power Systems*. 5.3 (August

[31] Huang, G. M., and N. C. Nair. "Detection of dynamic voltage collapse. *IEEE Power Engineering Society Summer Meeting*. 3 (2002): 1284-1289.

[32] Hydro-Quebec TransEnergie Technolgies,*SimPowerSystems Users Guide*, version. 3 (Sep. 2003).

[33] Huang Z., and J., Nieplocha, "Transforming power grid operations via High Performance Computing." *IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century* Pittsburgh, PA, (2008).

[34] IEEE/CIGRE Joint Task Force on Stability Terms and Definitions., "Definition and Classification of Power System Stability." *IEEE Transactions on Power Apparatus Systems*. 19.2 (May 2004): 1387-1401.

[35] IEEE/CIGRE Joint Task Force on Stability Terms and Definitions., "Interfacing Techniques for transient stability and electromagnetic transients program." *IEEE Transactions on Power Apparatus Systems*. 8.4 (October 2009): 2385-2395.

[36] IEEE Power System Relaying Committee, "Voltage collapse mitigation" (December 1996).

[37] IEEE task force report by the computer and analytical method subcommittee of the power systems engineering committee. "Parallel processing in power systems computation". *IEEE Transactions on Power Systems*. 7.2 (May 1992): 629 - 638.

[38] Ilic., M., M. L. Crow, and M. A. Pai., "Transient stability simulation by waveform relaxation methods". *IEEE Transactions on Power Systems*. 2.4 (November 1987): 943 - 952.

[39] Inabe, H., T. Futada, H. Horii, and K. Inomae, "Development of an instantaneous and phasor analysis combined type real-time digital power system simulator. *Proceedings of International Conference on Power System Transients*, New Orleans, LA, (2003).

[40] Jalili-Marandi V., and V. Dinavahi., "SIMD-based large scale transient stability simulation on the graphics processing unit". *IEEE Transactions on Power Systems*. 20.3 (August 2010): 1589 - 1599.

[41] Karypis, G., K. Schloegel, and V. Kumar., *PARMETIS Parallel Graph Partitioning and Sparse Matrix Ordering Library*. Department of Computer Science and Engineering, University of Minnesota, (August 2003).

[42] Kasztenny, B., and M. Kezunovic, "A method for linking different modeling techniques for accurate and efficient simulation. *IEEE Transactions on Power Systems*, 15.1, (February 2000): 65-72.

[43] Kasztenny, B., and M. Kezunovic., " A method for linking different modeling techniques for accurate and efficient simulation." *IEEE Transactions on Power Systems*. 15.1, (February 2000): 65-71.

[44] Kundur, P., *Power System Stability and Control*. New York: McGraw-Hill,1994.

[45] La Scala., M., M. Brucoli, F. Torelli, and M. Trovato., "A gauss-jacobi-block-newton method for parallel transient stability analysis". *IEEE Transactions on Power Systems*. 5.4 (November 1990): 1168 - 1177.

[46] La Scala., M., R. Sbrizzai, F. Torelli., "A pipelined-in-time parallel algorithm for transient stability analysis". *IEEE Transactions on Power Systems*. 6.2 (May 1991): 715 - 722.

[47] Lachs, W. R., and D. Sutanto. "Different types of voltage instability. *IEEE Transactions on Power Systems*, 9 (May 1994): 1126-1134.

[48] Lin, J., and J. R. Marti., "Implementation of the CDA procedure in the EMTP." *IEEE Transactions on Power Systems*. 5.2 (May 1990) 394-402.

[49] Liwei, W., D. Z. Fang, and T. S. Chung., "New techniques for enhancing accuracy of EMTP/TSP hybrid simulation algorithm. *Proceedings of IEEE International Conference on Electric Utility Deregulation, Restructuring and Power Technologies*, 2, (April 2004): 734-739.

[50] Makram., E. B., V. Zambrano, R. Harley, and J. Balda., "Three-phase modeling for transient stability of large scale unbalanced distribution systems". *IEEE Transactions on Power Systems*. 4.2 (May 1989): 487 - 493.

[51] Marti, J., and J. Lin., "Suppression of numerical oscillations in the EMTP". *IEEE Transactions on Power Systems* 4.2 (May 1989) 739-747.

[52] Milano, F., *Power System Modelling and Scripting*. London : Springer-Verlag., 2010.

[53] Perez, G., A. Flechsig, and V. Venkatasubramanian., "Modeling the protective system for power system dynamic analysis." *IEEE Transactions on Power Systems*. 9.4 (November 1994): 1963-1973.

[54] Siemens PTI Inc. *PSS/E Operation Program Manual: Volume 2* ver 30.2.

[55] Siemens PTI Inc. *PSS/E Application Guide, Volume 2* ver 30.2.

[56] *Power System Blockset users guide, version 1.* TEQSIM International Inc., 1999.

[57] Padilha., A., and A. Morealato., " A W-matrix methodology for solving sparse network equations on multiprocessor computers". *IEEE Transactions on Power Systems*. 7.3 (1992): 1023 - 1030.

[58] Reeve, J., and R. Adapa., "A new approach to dynamic analysis of ac networks incorporating detailed modeling of dc systems. Part I and II. *IEEE Transactions on Power Delivery*, 3.4, (October 1988): 2005-2019.

[59] Sauer, P.W., and M.A.Pai., *Power System Dynamics and Stability.*, New Jersey: Prentice Hall Inc., 1998.

[60] Semlyen, A., and M. R. Iravani., "Frequency domain modeling of external systems in an electro-magnetic transients program, *IEEE Transactions on Power Systems*. 8.2 (May 1993): 527-533.

[61] Shu J., W. Xue, and W. Zheng., "A parallel transient stability simulation for power systems". *IEEE Transactions on Power Systems*. 20.4 (November 2005): 1709 - 1717.

[62] Singh, H., and A. Abur., "Multi-port equivalencing of external systems for simulation of switching transients". *IEEE Transactions on Power Delivery*. 10.1(January 1995): 374-382.

[63] Smith, B., and H. Zhang, "Sparse Triangular Solves for ILU Revisited: Data Layout Crucial to Better Performance". *International Journal of High Performance Computing Applications*, 2010. DOI 10.1177/10 94342010384857.

[64] Strunz, K., R. Shintaku, and F. Gao., "Frequency-adaptive network modeling for integrative simulation of natural and envelope waveforms in power systems and circuits. *IEEE Transactions on Circuits and Systems*. 53.12 (December 2006): 2788-2803.

[65] Su, H. T., L. A. Snider, T. S. Chung, and D. Z. Fang., "Recent advancements in electromagnetic and electromechanical hybrid simulation. *Proceedings of the International Conference on Power System Technology.* (November 2004): 1479-1484.

[66] Su, H., K. W. Chan, L. A. Snider, and J. C. Soumagne., "Advancement on the integration of electromagnetic transients simulator and transient stability simulator, *Proceedings of International Conference on Power System Transients*, Montreal, (Jun. 2005).

[67] Su, H., K. K. W. Chan, and L. A. Snider., "Interfacing an electromagnetic SVC model into the transient stability simulation, *Proceedings of International Conference on Power System Technology*, 3, (Oct. 2002): 1568-1572.

[68] Su, H., L. A. Snider, K. W. Chan, and B. Zhou., "A new approach for integration of two distinct types of numerical simulator, *Proceedings of International Conference on Power System Transients*, New Orleans, (2003).

[69] Su, H. T., K. W. Chan, and L. A. Snider.,"Parallel interaction protocol for electromagnetic and electromechanical hybrid simulation. *Proceedings of the Institute of Electrical Engineers Conference*, 152.3 (May 2005): 406-414.

[70] Su, H. T., K. W. Chan, and L. A. Snider., "Investigation of the use of electromagnetic transient models for transient stability simulation. *Proceedings of 6th International Conference on Advances in Power System Control, Operation and Management*, (November 2003): 787-792.

[71] Sultan, M., J. Reeve, and R. Adapa., "Combined transient and dynamic analysis of HVDC and FACTS systems. *IEEE Transactions on Power Delivery* 13.4, (October 1998): 1271-1277.

[72] U.S. - Canada Power System Outage Task Force. *Final Report on the August 14th, 2003 Blackout in U.S. and Canada: Causes and Recommendations.*

[73] Vournas, C. D., G. A. Manos, J. Kabouris, and T. V. Cutsem. “Analysis of voltage instability incident in the Greek power system. *IEEE Power Engineering Society Winter Meeting.* 2(23-27 Jan 2000): 1483-1488.

[74] Wang, X., P. Wilson, and D. Woodford., “Interfacing transient stability program to EMTDC program., *Proceedings of IEEE International Conference on Power System Technology*, 2, (October 2002): 1264-1269.

[75] Wang, Y. P., and N. R. Watson., “Z-domain frequency-dependent A.C system equivalent for electromagnetic transient simulation. *Proceedings of IEE Conference on Generation, Transmission, and Distribution.* 16.1 (February 2001): 97-104.

[76] Watson N. R., “Improved fitting of z-domain frequency dependent equivalents for electromagnetic transients simulation. *Proceedings of Power Engineering Conference* (2007).

[77] Watson, N., and J. Arrillaga., *Power System Electromagnetic Transients Simulation.* London,UK: The Institution of Electrical Engineers, 2003.

[78] Wu., J. Q., and A. Bose., “Parallel solution of large sparse matrix equations and parallel power flow.” *IEEE Transactions on Power Systems.* 10.3 (August 1995): 1343 - 1349.

[79] Zimmerman, R. D., and C. E. Murillo-Sanchez., *MATPOWER 4.0 users manual.* Power Systems Engineering Research Center, (March 2011).