A Parallel Hybrid Method for Fast Analysis of Power System Dynamics

Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften
von der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte
Dissertation
von

M.Sc. Michael Kyesswa

Tag der mündlichen Prüfung:	11. Dezember 2020
Referent:	Prof. Dr. Veit Hagenmeyer
Korreferent:	Prof. Dr.-Ing. Albert Moser

Abstract

Computational simulations are currently extensively applied for analysis of power systems in order to ensure a secure and stable operation of the network. However, the actual trend in the power system operating environment shows several transformations in the grid structure as a result of increasing operation of large interconnected networks, growth in electricity demand, and the integration of renewable energy sources in the energy transition context. Such changes directly impose additional requirements to the stability analysis process, whereby the time-domain simulations widely used for dynamic stability studies are faced with an increase in computational burden due to the increasing complexity of the system under analysis. Nevertheless, the continuous changes in the system's operating point owing to variations in operation conditions shows a need for continuous analysis during network operation. This therefore necessitates advanced methods to cope with the introduced complexity in the analysis process.

The present thesis describes a new parallel hybrid computational method for the fast analysis of power system dynamics in large networks in order to address the above mentioned challenges. As a first step, mathematical models of power system components are described for representing the dynamic behavior of the system. New models of renewable energy sources – solar photovoltaic and wind power – are described for the functional representation of grid integrated distributed generation in system stability analysis. A phasor time-domain computational method is then presented for analyzing network dynamics in the electromechanical transient phenomena. In this case, a method is described for the conventional balanced transients' analysis and extended to a new method that includes analysis of unbalanced transients. To address the complexity in the analysis of large networks, a parallelization approach is proposed for the time-domain computations. For this, a grid partitioning strategy is presented for dividing networks into simplified parallelizable subnetworks, which are applied in a parallel-in-space computation approach. In a further step to account for continuous analysis including all scenarios in the analysis process, a direct-method is described for fast assessment of system dynamic stability. The direct-method introduces the ability of fast identification of critical network contingencies that require detailed analysis. With this property, the direct-method is combined with the parallel time-domain simulation approach to develop a hybrid computational method for fast, detailed and continuous analysis of power system stability during network operation.

The models and methods proposed in the present thesis are benchmarked against available open source and commercial software packages. As part of system integration, the methods are integrated into a modeling, simulation and visualization software framework which facilitates application of the methods in an interactive stability analysis environment. The results from the analysis in the integrated software framework show that the new methods provide an important contribution for setup of online power grid dynamic stability assessment in the smart grid context.

Acknowledgment

The work in the present dissertation was carried out under the Simulation and Visualization group at the Institute for Automation and Applied Informatics of Karlsruhe Institute of Technology. Writing this dissertation has been quite an experience and I am grateful to everyone who has supported me throughout the journey.

First of all, I would like to thank Prof. Dr. Veit Hagenmeyer in his capacity as the director of the Institute for Automation and Applied Informatics for granting me the opportunity to carry out my research at the institute. And more so, I am grateful to Prof. Dr. Veit Hagenmeyer as my main supervisor for the professional support and valuable counsel throughout the whole research period. Your continuous advice has greatly contributed to the successful completion of this dissertation. I would also like to extend my sincere thanks to Prof. Dr.-Ing. Albert Moser as the co-referee for reviewing and evaluating my dissertation.

Special thanks to Dr.-Ing. Uwe Kühnapfel the head of the Simulation and Visualization group for opening up the research opportunity under the group and for the support and constructive advice right from the start of my research journey. In the same way, I would like to express my gratitude to Dr.-Ing. Hüseyin Çakmak for the invaluable support, guidance and collaborative work throughout the research period. Working directly with you helped me to deepen my knowledge in the research field through the continuous exchange of ideas. I also acknowledge apl. Prof. Dr. Lutz Gröll for the mathematical insights that greatly contributed to the progress in the present thesis.

In addition, my sincere appreciation to my colleagues in the simulation and visualization group, Dr.-Ing. Heiko Maaß, Richard Jumar, and Anselm Erdmann for the wonderful working atmosphere and support in various forms that made the whole research experience worthwhile. I also thank my fellow Ph.D student Alexander Murray from the Optimization and Control group for the research collaboration which has contributed to part of the results presented in this thesis. Many thanks to the Energy Systems 2050 project of the Helmholtz Association for the financial support throughout my research period.

Finally, I extend my heart-felt gratitude to my family for the love, encouragement, guidance and unwavering support over the years that have led me to this achievement on my journey. I am forever indebted.

Karlsruhe, December 2020 *Michael Kyesswa*

Contents

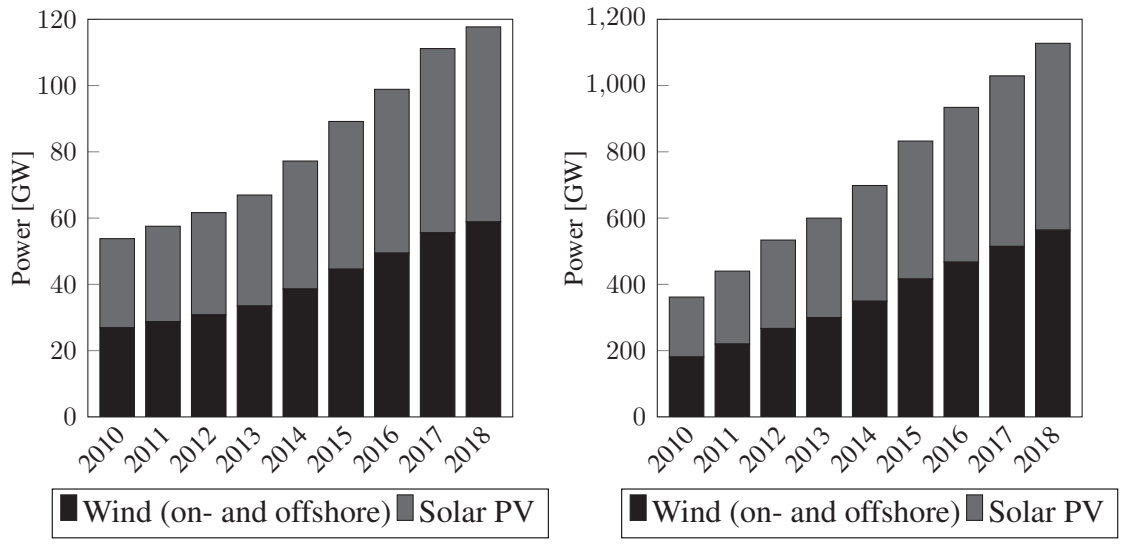
1 Introduction

1.1 Power Systems in the Context of Energy Transition

The electric power system is a central infrastructure supporting the operation of a number of critical systems. As a result, high requirements are placed on the security and stability of the electrical network in order to ensure continuous and reliable supply of electricity to end consumers. In traditional grids, network stability has been efficiently managed by the interconnected generation stations reacting to variations in network operating conditions to maintain the necessary balance between generation and demand. However, the need to address the climate change challenge necessitates solutions across different sectors to reduce greenhouse gas emissions [1]. The main contribution of the power system sector is the transition from dependence on conventional energy sources based on fossil fuels with a high level of carbon dioxide emission to alternative energy sources. Among the available alternative energy sources, variable renewable energy sources, especially solar photovoltaic (PV) and wind energy, are widely considered to have a great potential towards future low-carbon energy generation systems [2, 3]. This can also be observed from the actual trend, which shows a large growth in the installed capacity of such sources over the past few years. For instance, the trend in the installed capacity in Germany from 2010 to 2018 as depicted in Figure 1.1 shows a 151% increase in the solar PV capacity from 18.01 to 45.23 GW, and a 118% increase in the wind power capacity from 26.90 to 58.85 GW, considering both onshore and offshore wind power generation [4]. A similar trend in the installed electricity generation capacity is observed on the global scale as shown in Figure 1.1 [5, 6].

The increased integration of the variable renewable energy sources to meet the requirements of the energy transition [7] has led to transformations in the existing grid infrastructure. Firstly, from the power generation point of view, the primary energy sources are weather dependent and therefore variable and highly unpredictable [8]. This variability transforms into continuous mismatch between generation and consumption; thus affecting system stability in form of continuous frequency deviations such as the registered drop in the continental European grid frequency to 49.8 Hz in January 2019[1]. Secondly, variable renewable generators are mainly connected to the grid through power converter interfaces rather than using conventional synchronous machines. Synchronous machines are the main source of inertia, which contributes to system damping and thus determines the rate of change of frequency (RoCoF) following a disturbance. This capability is not inherently provided by the power electronic converters adapted for renewable energy sources [9]. Therefore, replacing conventional generators with more variable renewable generators results in a decrease in system inertia and may leave the power system vulnerable to sudden interruptions in form of disturbances. Thirdly, traditional

[1] https://gridradar.net/Unterfrequenzereignis_Januar_2019.html



(a) Germany scale

(b) Global scale

Figure 1.1: Net installed solar and wind electricity generation capacity from 2010 to 2018 [4]

power systems were constructed for unidirectional flow of power from generation, through the transmission and distribution networks, to consumers. Integration of distributed generation introduces a shift to bidirectional power flows with additional flows from the consumption points into the distribution and transmission networks. Furthermore, the connection points of variable renewable generators on the distribution level [10] result in regulation challenges for distribution system operators in maintaining the voltage profile within the standard acceptable operating limits.

A number of solutions have been proposed to address the challenges faced in the current grids in the process of achieving the energy transition. Among the suggested solutions are the flexibilities in energy storage systems, demand side management, and pooling of intelligent devices to form virtual power plants [8, 11]. The combination of distributed generation with such flexibility solutions in network operations has led to increased integration of information and communications technology (ICT) to interconnect the network intelligent devices. This integrated concept is referred to as a smart grid [12]. Several research projects have been initiated to study the requirements for future smart grid operations and devise possible solutions for the challenges associated with the energy transition [13]. Among such projects are the Energy Lab 2.0 demonstrator project [14] and Energy System 2050 (ES2050) research project[2] of the Helmholtz program.

In the current systems, detailed system studies are applied to ensure continuous network operation and reliable transmission and distribution of electrical power. Since system wide experimental setups are impractical for power system studies, computational simulations provide an alternative platform to perform the relevant tests and analysis studies as an integral part in the design, planning and operation of electrical networks [15]. The transformations observed in the

[2] https://www.helmholtz.de/en/research/energy/energy_system_2050

power grid structure as highlighted above, however, result in major changes in the power system operation environment. Network operators are continuously faced with new requirements for stability and dynamic performance of the integrated network system. The main requirement is to accurately represent the introduced transformations in the smart grid context in the analysis environment for studying the critical system security measures, such as frequency stability, voltage stability, rotor angle stability, and n-1 network security. In addition, there is a need to address the resulting increase in complexity involved in the continuous analysis of system stability and reliability during network operation. It is therefore of interest to advance power system models and simulation methods for the implementation of the smart grid solutions in order to cope with the changing analysis requirements. To this end, the current thesis presents new computational methods for transient stability analysis based on parallel time-domain simulations and direct stability analysis to address the challenges and increasing complexity in system analysis due to the changing power system operating environment.

1.2 Overview of Power System Simulations

Power system simulations are reviewed in the following to cover all stages of the electrical power system including system planning, design, development, operation and control. For power system operation, simulations aid in assessing the impact of potential network contingencies and provide information on variations in system operation after the occurrence of disturbances. This information is used by operators to identify possible strategies and solutions to mitigate the subsequent effects of network disturbances. In addition, power system operators rely on real-time, fast and accurate simulations to train network operators, assess the dynamic security of the network and plan short-term operations. From the system design perspective, computational simulations are essential in evaluating the impact of proposed system changes. This property is vital in the current power system environment which includes numerous changes, such as network expansions, integration of distributed energy sources, decommissioning of conventional power plants, and re-dimensioning of control and protection schemes for the new structures in the smart grids context. The various roles of simulation tools in the operation of power systems are summarized in Figure 1.2 [16].

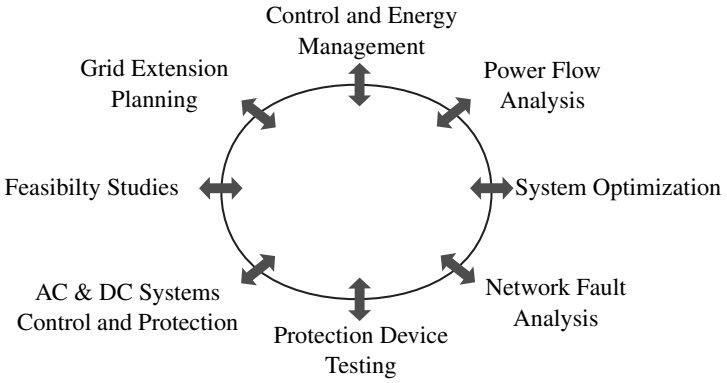


Figure 1.2: Roles of simulation tools [16]

The base computational simulation in power system studies is the power flow analysis [17], which evaluates the equilibrium operating states of the system under the assumption of steady state operating conditions. Power flow studies have been extensively used for predictive analysis of power systems in steady-state operation for any change in the system operating point [18]. Performing such analyses provides an overview of the system operation and is used to derive recommendations regarding control settings to optimize operation costs, while maximizing the grid transfer capacity. The main system variables analyzed are voltage magnitudes and phase angles at all buses, and the active and reactive power flows in the network transmission line sections.

However, the steady state assumption applied in power flow studies does not accurately represent the practical state of the power system which is continuously faced with random occurrences, like changes in load demand, network faults, equipment malfunction, sudden application or removal of loads, line failure, and loss of generation. Such occurrences cause significant system imbalances at all times resulting in a change in system operation from one equilibrium state to another. The network transients during the transition period may result in loss of synchronism or growing oscillations in the entire system. Dynamic simulations are therefore applied to analyze the system response to different disturbances and determine time-series responses during the transient period following a disturbance.

The possible time-scale separation of the power system dynamic behavior results in the division of dynamic simulations broadly into the following groups: transient stability simulations (TS) and electromagnetic transient simulations (EMT). Transient stability simulations are used to analyze electromechanical dynamics representing the interaction between the electrical and mechanical energies in the machines, which cause oscillations in the rotating masses following a disturbance. The time scale of electromechanical dynamics ranges from milliseconds to seconds [19]. The most significant assumption made during power system modeling for transient stability simulations is that the system frequency remains nearly constant, which implies that the voltages and currents can be expressed as fundamental frequency phasor quantities [20]. This significantly improves the computational efficiency in transient simulations by allowing a large time-step to be used during numerical computations. On the other hand, electromagnetic transient simulations are used to analyze dynamics caused by the interaction between electrical machines and the network. These transients constitute frequencies above the fundamental frequency and require a high level of precision for analysis [21]. For electromagnetic transient analysis, the system is modeled using the full three phase model by considering all phases, and the voltages and currents are represented by their exact waveforms allowing detailed analysis in time-domain to be conducted. The main difference between the numerical computation in transient stability simulations and electromagnetic transient simulations is the discretization time step. Electromagnetic transient simulations require a smaller time step, in the range of micro seconds or less, in order to capture the fast dynamics involved with such transients. The small time step and detailed modeling result in higher computational complexity in electromagnetic transient simulations compared to transient stability simulations. The next section describes the developments in power system simulations.

1.3 State-of-the-Art in Power System Computational Methods

Computational simulations have been developed for system wide studies considering steady-state and dynamic conditions. Specific to dynamics simulation as the focus of the present thesis, a number of simulation methods exist for various applications with varying levels of complexity and analysis accuracy. The current section presents the state-of-the-art of the methods and software packages applied for the power system transient stability analysis in research and industry. Section 1.3.1 describes the main analysis methods, based on time-domain simulations, and the development efforts which have been undertaken to include asymmetrical component modeling in system analysis. Since the solution methods applied in time-domain simulations result in high computational requirements for analyzing large networks, Section 1.3.2 describes the applied parallelization techniques to improve the computational efficiency in time-domain simulations. In the context of dynamic security assessment, the developments in the alternative transient stability analysis approaches based on direct methods are presented in Section 1.3.3.

1.3.1 Time-Domain Simulation Approach

In research and industry, the computational methods applied for power system stability analysis mostly rely on the time-domain simulation approach [22]. This approach is based on the step-by-step numerical solution of the differential and algebraic equations describing the system dynamic behavior and its interaction with interconnected subsystems over different time scales. The main advantage of using time-domain simulations is that they can be applied to any level of detail of power system models. This provides the benefit of directly applying detailed models to accurately represent the system components during simulations. It has been observed that the higher the model details, the better the representation of the transient behavior in the whole system [23]. It is for this reason, combined with the accurate numerical integration applied in the solution approach, that time-domain simulations achieve a high level of simulation accuracy.

The existing computational software packages applying time-domain simulations are broadly divided into two groups: commercial and open source simulation tools. The available commercial software packages have proved to be highly reliable, computationally efficient and user-friendly, with a wide range of available models and functionality; including electromagnetic and transient stability analysis. In view of the separation in time-scales, separate software packages have been developed for transient stability (TS) and electromagnetic transient (EMT) simulations with a different level of modeling complexity and computational accuracy. The software packages for electromagnetic transient analysis are used for detailed modeling and analysis of dynamic conditions where the main requirement is high simulation accuracy and consideration of fast transients, network nonlinearities and unbalanced conditions [21]. Examples of commercial tools for electromagnetic transient analysis include EMTP-ATP [24], PSCAD/EMTDC [25], RSCAD/RTDS [26], and SimPowerSystems [27]. Very small time steps are applied in such tools for discretization of the differential equations in the numerical computation process in order to capture the very fast dynamics. The characteristics of very small time steps together with the detailed component modeling result in high computational requirements in electromag-

netic transient simulations. It is therefore computationally challenging to apply EMT tools in analyses involving large networks; thereby limiting their application to relatively small network sections.

Simplifying assumptions are considered during component modeling in commercial packages with transient stability analysis modules such as PSS/E [28], DIgSILENT PowerFactory [29], and NEPLAN [30]. The result of such simplification is a relaxation in the maximum allowable simulation step size to make the overall computation as time-efficient as possible. It is due to this reason that these tools are extensively used for stability analysis considering system wide transient behavior in large networks. From a research and educational perspective, the main drawback of commercial software packages is that they have a closed architecture [17]. This implies that users have no access to the source code to explore the implementation aspects in order to modify system models. Users are mainly restricted to the level of accepting the assumptions and simplifications considered during the development phase of the software packages. This in turn results in a limitation on the ability to extend the functionality and scalability of the tools to address the changing network requirements.

In view of the limitations in commercial software packages, open source simulation tools have been developed for research and educational purposes. Open source tools provide users with easy access to source code, which allows modeling flexibility and ease of functionality extension. The benefit of modeling and functional flexibility is that the challenges arising due to the changing power system operating environment in the smart grid context can be easily investigated through continuous development and improvement of the power system analysis methods. Table 1.1 shows examples of existing open-source static and dynamic simulation tools for power system analysis. The analysis functions shown in Table 1.1 are power flow (PF), optimal power flow (OPF) and time-domain simulation (TDS). An extensive description of the open source software packages with different levels of complexity and analysis functionality is given in [17, 31].

The key simplifying assumption in transient stability analysis is the consideration of balanced three-phase operating conditions in order to reduce the size of the system to be solved. In this case, the complexity of the system model is reduced by considering only a single phase positive sequence network during the analysis. The discrepancies that exist between the three phases are disregarded in this kind of analysis. This implies that the errors caused by neglecting the differences in the magnitudes of the voltage and the phase angle between the three phases are considered to be negligible. For operating conditions – such as due to unbalanced loads and during asymmetrical faults which form the majority of fault types in a real power system – where system imbalances cannot be ignored, it becomes increasingly relevant to include the analysis of asymmetrical transients.

Major steps have been taken in research and industrial applications to extend power system analysis to asymmetrical system studies. The significant developments have mainly focused on developing individual component models, applying modeling and analysis techniques specific for asymmetrical systems, and combining the models and analysis methods for application to the different categories in power system stability studies. Specific component models have been developed for transformers [45] and synchronous generators in the sequence and phase coordinates for application to large network studies [46, 47]. The main method applied in the

Table 1.1: Examples of open-source simulation tools for power system analysis

Analysis domain	Tools	Language	Analysis functions		
			PF	OPF	TDS
Static simulation	GridLAB-D [32]	C/C++	✓		
	Matpower [33]	Matlab	✓	✓	
	PyPower [34]	Python	✓	✓	
	PyPSA [35]	Python	✓	✓	
Dynamic simulation	PSAT [36]	Matlab	✓	✓	✓
	InterPSS [37]	Java	✓		✓
	OpenModelica [38]	Modelica			✓
	OpenDSS [39]	Delphi	✓		✓
	MatDyn [40]	Matlab	✓		✓
	GridCal [41]	Python	✓		✓
	Dome [42]	Python	✓	✓	✓
	pypower-dynamics [43]	Python			✓
	Pandapower [44]	Python	✓	✓	✓

literature for asymmetrical power system analysis in time-domain simulations is the symmetrical components technique. This method decomposes the asymmetrical three-phase system into three decoupled sequence circuits that can be analyzed on a single phase basis [48, 49]. The symmetrical components method has been applied for unbalanced three-phase power flow simulation studies, such as considering formulation of generator, transformer and transmission line models presented in [50], applied to micro-grids and active distribution system as described in [51], and for medium and low voltage networks including distributed generation from photovoltaic generators in [52]. Applications of the symmetrical components method for system stability assessment include: small signal stability assessment (SSSA) [53], dynamic analysis for fault studies in distribution and transmission networks [54] and a combination of three-phase unbalanced transient dynamics and power flow simulation studies for micro-grid and distribution-level modeling [55].

Another method for modeling and analysis involving asymmetrical systems is based on dynamic phasors [56]. This method incorporates frequency information in the dynamic modeling of the power system components. In the dynamic phasor formulation, the fast varying time-domain instantaneous quantities are represented as a series of slow varying Fourier coefficients of a specific time window of interest [57]. The dynamic phasor approach has been applied for asymmetrical fault analysis, analysis of unbalanced polyphase machines and power electronic modeling as evaluated in [58].

The above approaches focused on developing methods for separately analyzing transient stability and electromagnetic transients. In order to close the separation gap between EMT and TS analysis, further research efforts have focused on extending the analysis methods to hybrid

simulations and co-simulation platforms (for an overview cf. [20]). The aim in such methods is to combine the computational efficiency in transient stability simulations with the accuracy of electromagnetic transient simulations. Thereby, a small part of the power system is simulated using the accurate electromagnetic transient program, while the rest of the system is modeled by the computationally efficient transient stability programs. In hybrid simulations, the TS and EMT approaches are implicitly coupled to combine and simultaneously solve the equations of the two phenomena as presented in [59, 60]. On the other hand, co-simulation platforms use an explicit approach to interface separate tools by running the two programs independently and exchanging network information via defined data exchange modules at regular time intervals as described in the applications presented in [61, 62, 63, 64, 65]. Hybrid and co-simulation methods have been extended to applications integrating the simulation of transmission and distribution network dynamics. Further details about such applications can be found in [66, 67, 68, 69].

1.3.2 Parallel Time-Domain Simulations

The simulation process in the time-domain approach is computationally demanding due to the complexity of the numerical integrations involved in the solution of the large system of equations resulting from modeling the power system [70]. Parallel processing techniques, combined with improved solution algorithms, have been proposed to achieve better computational performance in power system simulations [71, 72]. The increasing need for parallel solutions for power system simulations is further supported by the advances in computational technology shifting from running applications on single computers – with a single processor – to a distributed and parallel computing architecture. A number of methods and applications of high performance computing in power system studies have been reported in the literature [73, 74, 75]. Specific to dynamics simulation, parallel computing techniques depend on the decomposition of the system's differential and algebraic equations into subsystems that can be parallelized based on the parallel-in-space, parallel-in-time, and waveform relaxation algorithms [72].

In the parallel-in-space algorithm, the network is partitioned into independent subnetworks and the subnetwork equations are assigned to different processors. One approach for the parallel-in-space network decomposition is the Block-Bordered Diagonal Form (BBDF) [76], which splits the network based on node sectioning. This results in division of the matrix representing the network interconnections into a form where the subsystems can be solved in parallel. The information exchange points in this formulation are the interface nodes between the subsystems, through the interface node voltages. The BBDF network decomposition approach been has applied to implement transient stability simulations on parallel and distributed computation platforms as described in [77, 78, 79]. Another approach for the parallel-in-space decomposition is the Multi-Area Thévenin Equivalent algorithm (MATE) [80], based on branch sectioning. In the formulation of the network subsystems, each end of the sectioned network branch is represented by a Thévenin equivalent in the adjacent subsystem. The interconnecting branch currents are used for the global information exchanged between subsystems. A similar branch sectioning strategy has been applied for the implementation of parallel dynamic simulations in [81] and [82].

The parallel-in-time algorithm is applied by combining the differential and algebraic equations over several time steps to create a larger system which can then be solved simultaneously. The equations over several integration steps are assigned to each processor. An example of such an algorithm is the parallel in-time block-Newton method described in [83]. The approach uses the trapezoidal integration method for discretizing the differential equations and formulates the problem for simultaneous solution for all time steps. Other applications of the parallel-in-time algorithm for transient stability analysis are reported in [84, 85].

The waveform relaxation algorithm was proposed in [86, 87, 88] for transient stability analysis. The method is based on domain decomposition in which the differential algebraic equations of the system are separated into subsystems and distributed to different processors to be solved simultaneously. For the simultaneous execution to derive the combined system response, the different subsystems approximate the behavior of the neighboring subsystems using previously estimated variables. The solution in each subsystem is reduced to solving for unknowns specific to that subsystem. Variables are exchanged between subsystems to update the estimates in the other subsystems until convergence is achieved. The waveform relaxation method has been extended and implemented in parallel applications using shared memory computers [89], in real time transient stability analysis [90], and hybrid coupled electromechanical and electromagnetic transient simulations [91].

An overview of the applications of high performance computing in power systems and trends in algorithms and computing hardware is given in [74]. Further implementations and advances in the use of parallel and distributed computing techniques for efficient computations in power and energy system simulations are described in [92].

1.3.3 Direct Analysis Approach

The analysis methods described in Section 1.3.1 and Section 1.3.2 are based on the time-domain simulation approach. Time-domain simulations generally provide satisfactory system analyses, but the computational complexity involved in the applied solution techniques based on numerical integrations limits their application to a limited set of test operation scenarios. A complete power system stability analysis, however, usually requires several studies in order to analyze the network response to all the possible disturbances and account for the continuous variation in power system operation conditions. This is specifically important in the smart grid environment, which experiences continuous changes in the operating point due to the variable nature of generation levels from the renewable energy sources. An alternative analytical approach for transient stability assessment is provided by the direct analysis methods. In general, direct methods assess system stability using algorithmic techniques by comparing the system energies during the fault period and at the initial post-fault state. The computational advantage in direct methods over time-domain simulations during stability estimation is achieved by eliminating the need for the time consuming numerical integration of the system differential equations in the post-fault period [93].

Direct methods are divided into two groups based on the characteristic assessment technique: the equal area criterion based methods, and the Lyapunov-based methods. An overview of the state of the art of direct methods is given in [94]. The equal area criterion (EAC) based

methods apply a graphical assessment approach for the transient stability analysis. System stability is derived directly by comparing the energy gained during fault and the energy that can be absorbed in the post-fault period. A detailed description of the equal area criterion is given in [95, 96].

In the earlier applications, the EAC method was limited to one-machine infinite bus (OMIB) and two-machine systems. The EAC has been extended to the analysis of multi-machine systems using an approach referred to as the extended equal area criterion (EEAC). The analysis in the EEAC is based on an aggregation strategy of a multi-machine system into one set of critical machines and another of non-critical machines to form a two-machine system. The formed two-machine system is then analyzed using the basic equal area criterion. The formulation and analysis procedure in the EEAC method is described in [95]. The extended equal area criterion has been applied for contingency filtering in transient stability analysis processes as described in [97]. The equal area criterion method has further been applied to implementation of a hybrid approach known as the single machine equivalent (SIME). The SIME approach combines time-domain simulations with the equal area criterion for the effective screening of contingencies. The idea of the SIME method is to apply time-domain simulations to a multi-machine system in order to derive an equivalent two machine system, which is then analyzed using the equal area criterion [98, 99]. The SIME method has been extended to applications in contingency and stability assessment studies as described in [100, 101].

Transient stability assessment using Lyapunov-based direct methods [102] relies on definition of a transient energy function for a given power system under analysis. The system stability is analyzed using the system trajectory, and a specified region of attraction. The general analysis approach in the Lyapunov-based methods can be summarized in two main steps: The first step computes the relationship between a pre-fault stable equilibrium point and the system state during fault. To achieve this, numerical integrations are carried out on the system equations during the fault period to derive the fault-on trajectory. The integration process is continued until the fault clearing time. The second step of the analysis directly derives the system stability by analyzing the location of a given state in relation to the region of attraction. If the initial state of the post-fault system lies inside the stability region of a specified stable equilibrium point, then it can be concluded that the system will return to the equilibrium point. This is carried out without numerical integration of the post-fault system equations.

The main requirement in the application of the Lyapunov-based direct methods is the approximation of the region of attraction. The following energy-based methods have been developed for the estimation of the region of stability as described in [102]: Closest unstable equilibrium point method (Closest UEP); Controlling unstable equilibrium point method (CUEP); Boundary of stability region-based Controlling UEP method (BCU); and Potential energy boundary surface (PEBS). The difference in these approaches is in the calculation of the critical energy, which is used to estimate the stability region boundary in power system applications [102]. The first three approaches require a detailed computation of the unstable equilibrium points (UEPs), in order to determine the closest or controlling boundary equilibrium point which is used to estimate the critical energy. In power system applications using direct methods, computing the UEPs is the most computationally demanding step in the analysis process. On the other hand, the PEBS method eliminates the need of computing the UEPs by considering an approximation

using the potential energy, which results in computational simplification. However, it is shown in the literature that the existing PEBS methods based on maximum potential energy [103] and the directional derivative approximation [104] generally give conservative approximations as compared to the BCU method [102].

The different Lyapunov-based direct methods are analyzed in [105] for the application to dynamic studies in large-scale electric power systems. Additional applications of the Lyapunov-based direct methods are shown in the implementations of fast and accurate stability control strategies using the PEBS method [106], and application of the CUEP method in transient stability screening and contingency categorization [107]. Furthermore, a comparison of the Lyapunov-based methods and equal area criterion based methods is given in [108] in an effort to assess the adaptability of the methods for real time operation. According to the study, the efficiency of the Lyapunov-based methods is related to the network reduction technique applied during the solution. On the other hand, the efficiency of the equal area criterion based SIME method increases as the ratio of the generation nodes to total network nodes increases. Recent applications have shown a renewed interest in direct methods for transient stability assessment, as seen in the application of the general Lyapunov method in [109], improvement in estimations using the CUEP method presented in [110], and the revived application and improvement of the SIME assessment approach in [111, 112].

As far as dynamic security assessment is concerned, the complete assessment of power system stability usually involves a combination of techniques using power flow analysis, time-domain simulations and direct methods to achieve efficient computations with an acceptable level of accuracy. For this purpose, hybrid methods have been developed to combine the advantages of direct methods and time-domain simulations for dynamic security assessment in power system operation. The hybrid methods reported in the literature are based on the SIME-method and the BCU-method as the direct method components connected to standard time-domain simulation tools. Further information regarding the state-of-the-art of the tools and analysis methods applied in industry for the different aspects involved in dynamic security assessment can be found in [22].

1.4 Open Questions

Despite the extensive developments in methods and tools for power system stability analysis identified in the literature, the continuously changing power system operating environment creates new challenges for the system dynamics analysis problem. Therefore, further research is required in the modeling and development of simulation methods to address the following open questions:

- Flexibility in analysis tools: Time domain simulations are widely applied for power system operations with acceptable accuracy levels for network simulation studies. The current simulation tools with extensive analysis features are the commercial software packages that are mainly used in industry. Such tools, however, are observed to offer less flexibility in the research and education environment in terms of modification of system models, extension of analysis functionalities, and scalability of computation methods.

In order to facilitate further research into possible solutions to address the continuously changing requirements in the power system operation environment, extensive flexibility is required for research-based analysis software packages.

- Model extension: The difference in time scales of the component dynamics has been addressed by applying simplified models in the transient stability analysis function; thus reducing the computational effort. However, more accurate system analysis depends on the details of all the component models that affect the dynamic behavior of the system. Therefore, it is necessary to revise the models used in the simulation tools to improve the dynamic representation of the system components.

- Functionality extension: The main simplification in transient stability analysis tools shown in Table 1.1 is the consideration of balanced network operating conditions to reduce the computational complexity. However, this condition does not correctly represent the practical power system state, which is continuously faced with network imbalances especially in the current grids with distributed energy sources. Regardless of the existence of models and methods for asymmetrical system analysis, the methods in research have been developed for individual components and separate analysis functions which are tested in commercial software packages. It is therefore necessary for research-based software packages to include the asymmetrical transient analysis function to account for a wide range of transients in system studies.

- Modeling of renewable energy sources: The development of models to represent renewable energy sources in the current power system has been driven by standard grid codes. Various studies have been undertaken to develop dynamic models for studying the behavior of the system in response to the increase in the new generation sources. However, there is still a lack of validated models especially for renewable energy sources connected at the distribution level, which form the majority of the sources. This can also be seen in the large number of studies that ignore the dynamic behavior of renewable energy sources by applying the simplified negative load model during the analysis [10]. Therefore, it is of interest to derive models that accurately represent the dynamic behavior of grid connected renewable energy generators based on grid operation codes.

- Computational complexity: The changes seen in the current grids in light of the increasing operation of large interconnected networks, growth in electricity demand from e.g. electric vehicles and heat pumps, and integration of renewable energy sources result in an increase in complexity of the systems under analysis. This transforms into an increase in computational complexity of the transient stability analysis process. The approaches proposed in the literature for improving the analysis efficiency are based on parallel computation using network decomposition schemes and numerical solutions using implicit integration methods. However, these approaches have not been applied to extend the optimized and efficient sequential algorithms to parallel computations. In this regard, improved computation algorithms are required using the fundamental parallelization techniques and state-of-the-art high performance computing environments in order to address

computational burden in the simulation and analysis process. In addition, since the fundamental parallel approaches rely on system decomposition, it is important to derive optimal network decomposition strategies in the formulation of the parallel solutions.

- Continuous system analysis: The complete analysis in power systems requires continuously analyzing a large set of contingencies during network operation. Despite the computational efficiency in time-domain simulations used for the accurate analysis of transient stability, the overall computational effort involved usually limits their application to a few analysis scenarios. The alternative approaches based on direct methods generally provide fast analysis. However, analyzing the performance of the direct methods in the literature shows that there is a requirement to balance the analysis accuracy and efficiency in the methods. This requires a method which eliminates the challenging step of computing the unstable equilibrium points during the analysis and reduces the conservativeness in stability assessment. Moreover, since the direct methods in general do not provide complete analysis details, a combined approach with time-domain simulations is necessary for the complete system analysis, providing fast derivation of system stability state and detailed analysis of the system transient behavior.

- Validation of models and methods: Testing of component models and methods is an important step in the validation of the applicability of the developed methods in system analysis. It is also necessary to benchmark and evaluate the developed solutions for the correct functionality in the changing power system operating environment using standard and practical networks.

- Testing in smart grid environment: The identified power system operation is tending towards smart grid operations and several solutions are being developed for smart grid applications. However, it is necessary to establish how the new solutions fit into the smart grid operating environment. To achieve this, it is of interest to integrate component models and solution methods in testing platforms that represent the future smart grid environment.

1.5 Objectives

The motivation of the present thesis is the need to address the open questions and challenges identified within the existing analysis methods and to account for the changing requirements in the power system operating environment during stability studies. The main objective of the thesis is therefore to develop new computational simulation methods for fast analysis of power system dynamics in large networks. The general concept in the development is to consider enhanced time-domain simulations with extensive computational features for balanced and unbalanced network transients' analysis, combined with a direct analysis approach for fast derivation of the stability state under different network configuration during the system analysis process. The implementation goal in the present thesis is an extendable and scalable research-based hybrid computational method combining a direct-method and parallel time-domain simulations.

The specific work packages executed in order to achieve the thesis objective are described in the following subsections.

1.5.1 Mathematical Modeling

More accurate representation of the power system and its components is necessary for a comprehensive study of the dynamic behavior of the system. In computational simulation methods, the system representation is defined in terms of mathematical models. For this reason, mathematical models are derived to describe the fundamental components constituting the power system in the present thesis. Generally, the details of the mathematical models used for power system components are related to the time-scale separation of the simulation functions, which are categorized into steady state, transient stability and electromagnetic transient analysis. The modeling in the present thesis is specifically for steady state and transient stability analysis phenomena. Component models are developed for the analysis of transients in the following computational functions:

- Transient stability analysis function (TS): In this simulation function, the power system component models are represented in a symmetrical and steady state form. Balanced three-phase networks and operating conditions are assumed, which simplifies the modeling by considering the network model on a per-phase basis. The system frequency is assumed to remain constant, and voltages and currents are expressed as fundamental phasor quantities. The models are used for studying slow dynamics and electromechanical transients under balanced conditions. The computational method using this kind of modeling is referred to as the symmetrical transient stability analysis method in the present thesis.

- Three-phase transient stability analysis function (3ph-TS): Network components in this function are represented on a steady state three-phase basis. In addition to the balanced conditions analyzed using the TS function models, unbalanced operating conditions are also considered and system modeling is undertaken using the symmetrical components technique where the three sequence networks (positive-, negative-, and zero-sequence networks) are applied. The system frequency is assumed to remain constant and voltages and currents expressed as fundamental phasor quantities. The simulation function is applied for studying electromechanical transients resulting from balanced and unbalanced network disturbances. The computational method applying such models is referred to as the three-phase transient stability analysis method in the present thesis.

The major change in terms of structural composition in the current power systems is the integration of renewable energy sources and replacement of the conventional generation sources. To represent this transition, dynamic models of renewable energy sources – solar photovoltaic and wind power – are derived for studying the effects of the integration of distributed energy generation based on standard grid operation codes.

1.5.2 Development of Computational Methods

The main aspect in the present thesis is the development of new computational simulation methods for the analysis of power system dynamics. Computational solution methods are derived for the respective transient analysis functions described in Section 1.5.1. The methods are combined into a flexible and extendable phasor time-domain simulation toolbox for the analysis of balanced and unbalanced network transients. In light of the high requirement on computational performance in the analysis process, additional measures are derived to improve the efficiency of the computational methods. This is carried out considering individual simulation cases and the complete system stability analysis under different operating conditions. The following computation advancements in terms of environment, technique and method are applied to address the network complexity and improve the computational efficiency:

- Computation environment: This approach considers the advancement of the derived simulation methods using the same solution technique but in a computing environment which offers high performance computing capability. For this, the implementation of the computation method is transferred from a Matlab-based environment to a Julia-based computing environment [113].

- Computation technique: In this approach, the power system problem is reformulated in such a way that the solution of the resulting system of equations can be executed in parallel. A new parallel solution technique is therefore developed, based on a parallel-in-space scheme and an extended graph partitioning strategy, to adapt the time-domain computational method to high performance computing technology. This allows the use of parallel and distributed computing mechanisms to take full advantage of multi-core processors and cluster computing in order to speed-up the time-domain simulations.

- Computation method: This approach addresses the need to speed-up the whole analysis process, which involves studying a large set of network contingencies. Thereby, an alternative stability analysis approach based on a direct-method is developed for fast stability analysis. As a result, a computational method combining the direct-method and parallel time-domain simulations is developed for the complete and detailed analysis of system stability considering all network disturbances and the variation in network operating conditions. This method is referred to as the hybrid computation method.

Figure 1.3 illustrates an overview of the development concept, showing the developed analysis functions for time-domain simulations and the enhancement in analysis efficiency across the computation method and parallel computation technique.

1.5.3 Evaluation and Benchmarking

The developed models and computational methods are tested to validate their functionality. The evaluation in the development process is carried out by comparing the developed models and algorithms against open source and commercial simulation software packages at each stage. This is undertaken to assess the relative accuracy of the component mathematical models and

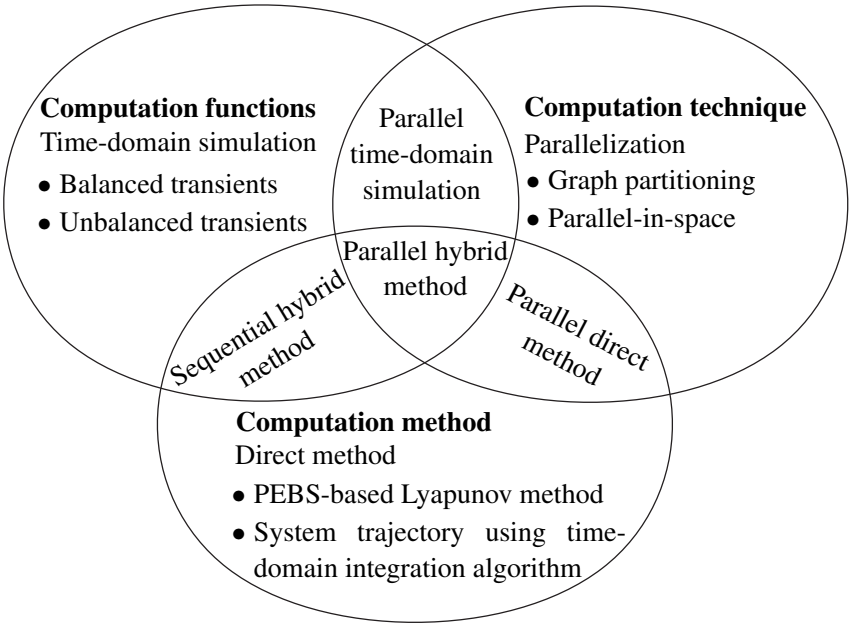


Figure 1.3: Overview of the combined analysis framework

algorithms in relation to validated models in existing tools in a process of benchmarking. Standard test network structures – such as IEEE test networks and open-source networks – and real networks are used for benchmarking purposes.

1.5.4 System Integration

The computational simulation methods developed in the present thesis are further integrated into the *e*ASiMOV (for energy system Analysis, Simulation, Modeling, Optimization, and Visualization) software framework [114, 115]. The framework is thus extended for interactive modeling, editing, graphical representation and interactive analysis and visualization of the dynamic behavior in complex power system networks. The integrated framework contributes to the EnergyLab 2.0 project [14], which provides a facility with software and hardware settings for studying the interaction between components in future energy systems and for testing new solutions necessary for the energy transition.

1.6 Thesis Outline

In Chapter 2, the mathematical modeling of the conventional power system components is described. Firstly, component models applied in the balanced system analyses and the corresponding simplifying assumptions are presented. To address the requirement of extending the analysis functionality, the component models derived for unbalanced system analyses are described. The aim of the chapter is to provide a clear definition of the application domain of the derived models and an understanding of the mathematical representation of the power system and the interaction between the different components in the whole system.

Chapter 3 describes the developed time-domain simulation method for transients' analysis in the conventional power system. Initially, the fundamental building block of the time-domain computational method is described based on an open-source simulation framework for analysis of balanced power systems. This is followed by the extension of the method to unbalanced system analyses; thus describing a new simulation module for the analysis of balanced and unbalanced power system transients.

Following the modeling and analysis in conventional power systems, Chapter 4 further describes the integration of renewable energy sources in stability studies of power systems. For this, the chapter describes the mathematical modeling of solar photovoltaic and wind power generation systems. The simplifying assumptions in the modeling process are described to define the analysis domain of the integrated system. The developed models are integrated into the time-domain simulation methods developed in Chapter 3 and tested in the integrated power system for compliance with standard grid operation codes.

Chapter 5 describes the advancement of the time-domain simulation method in terms of computation environment and computation technique using a parallelization scheme. The main focus in the chapter is the description of the developed parallel computation algorithm for power system transient stability analysis and assessment of its performance in terms of accuracy and computation speedup.

In Chapter 6, the third measure in advancement of the performance of the developed computational algorithms in terms of computation method is addressed. For this, the chapter describes the development of an alternative analysis approach based on a direct-method for fast stability assessment.

The aim of Chapter 7 is to describe the formulation of the parallel hybrid analysis approach coupling the direct-method and time-domain simulation method to create an analysis approach for fast and detailed analysis of transient stability. In addition, the chapter describes the integration of the developed methods as simulation modules into the *e*ASiMOV framework, which was developed as part of the research and development work at the Institute for Automation and Applied Informatics of Karlsruhe Institute of Technology and contributes to the Energy Lab 2.0 testing facility.

Finally, Chapter 8 summarizes the present thesis, including the main contributions and outlook for possible future work.

2 Conventional Power System Modeling

Knowledge of the power system structure and its components is an important step in the development of computational methods for power system stability analysis. Mathematical modeling is a vital concept used to describe the system components and their effect on the transient behavior of the system. A comprehensive system model should include the steady state and dynamic representation of the various elements influencing the electrical and mechanical torques of the machines in the system in order to capture the different scales of dynamics. The complexity of the models applied in the computational methods, however, depends on the type of transients and the system under investigation. The current chapter describes the mathematical models of key components that constitute an electrical power system for electromechanical transient stability analysis. The models are described using a set of differential and algebraic equations. Generally, differential equations mainly model the behavior of the machines, while algebraic equations are used to represent the transmission network and the interface between the machines and the transmission network. In the current chapter, component models are initially described for the traditional transient stability analysis computational method based on balanced network operating conditions. Modified models are then developed for an extended computational method for analyzing asymmetrical network transients. Underlying assumptions are also described to define the basis for the complexity simplification in the developed computational models.

2.1 General Modeling Assumptions

The power system model considered in the current thesis is broadly divided into the generator subsystem and the network subsystem. The generator subsystem includes the synchronous generators and the associated controllers, such as the turbine-governor system and the excitation system. As part of the network subsystem are buses (also referred to as nodes), transformers, and the interconnecting transmission lines. System loads are included as part of the network subsystem. The interface between the two subsystems is through the stator of the generator. A complete representation of the components highlighted above results in a large number of differential and algebraic equations (DAE) that have to be solved during the computation process.

The modeling details presented in the current chapter are based on the publications [116, 117].

M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Generator Model Extension for Higher Accuracy Simulation of Power System Transients in OpenModelica," in *4^{th} International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, Corfu, Greece, pp. 44–50, August 2017.

M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Simulation Tool for the Analysis of Unsymmetrical Power System Transients in Large Networks," in *European Conference on Modelling and Simulation (ECMS)*, Wilhelmshaven, Germany, pp. 246–253, May 2018.

Moreover, analyzing the time scales of the different components shows that the power system is naturally a stiff system, consisting of a wide range of time varying dynamics associated with the various components. In view of the broad structural composition, the computational analysis of power system dynamics using detailed component models to capture multi-scale system transients is generally a challenging task.

In order to simplify the modeling and reduce the complexity in the analysis process, a number of assumptions are introduced in the models derived in the current thesis. The main assumption, as applied in transient stability computational software packages, is the quasi-steady state approximation of the network [96]. This assumption implies that the network quantities are considered to be at fundamental frequency, neglecting all harmonics, and represented by perfectly pure sinusoidal signals. The sinusoidal voltages and currents are further simplified to fundamental phasor quantities, represented using an amplitude, phase angle, and a fixed angular frequency [20]. This simplification is acceptable in transient stability studies since the analyzed transients are slow compared to the propagating signals. The implications of the quasi-steady state assumption on the dynamic behavior, network modeling and computational complexity are summarized as follows:

- Dynamic behavior: The interaction between the different dynamic components is considered to occur with the electrical network at a constant frequency. Therefore, the transient response to the system's machine dynamics is only represented using the time variations in the current- and voltage-phasors [96].

- Network modeling: By assuming instantaneous variations of the network voltages and currents, the transients in the line elements are ignored. The transmission line model is therefore approximately modeled using the simplified lumped line model [96] representation of the line behavior from the interconnecting terminals. In addition, the assumption of constant network frequency simplifies the electrical network representation to a phasor model using equivalent impedances or admittances instead of the elementary resistive, inductive and capacitive (R, L, C) components.

- Computational complexity: The phasor representation allows transient stability simulations to use a time step in the range of milliseconds, depending on the investigated phenomenon, which reduces the computational burden during the numerical solution of the system equations.

The above simplifications and assumptions result in considerable model reduction that allows the models to be effectively applied in the analysis of relatively large network structures. It should be noted that the model reduction under the above suppositions is achieved at a cost of reduced computational accuracy. This limits the application range of the models in the present thesis to slow system transients in the electromechanical domain. The fast transients in the electromagnetic domain that require detailed component modeling are not taken into consideration in the current thesis.

2.2 Component Models for Transient Stability Analysis

Transient stability analysis provides the basis in the simulation of power system dynamics based on the simplifications outlined in Section 2.1. In addition to the assumptions stated above, the conventional transient stability methods apply further simplification of the network in order to reduce the size of the network system equations. The simplification is reached by assuming balanced network and operating conditions in normal operation and during fault conditions [20]. In such a case, the three-phases of the electrical network are assumed to have uniform characteristics and the network dynamics can be analyzed by considering a single phase. As a result, the system representation is further simplified to a single-phase equivalent of the three-phases based on the positive sequence network. The current section describes the mathematical models of the fundamental system components applied for transient stability computations under balanced and symmetrical network conditions in the present thesis.

2.2.1 Synchronous Generator

Before describing the generator models, it is necessary to make a note on the sources of dynamics in the system and how they are represented. The synchronous generator is the main source of system dynamics as a result of the electromechanical and electromagnetic interactions between the mechanical and electrical components in the internal generator structure. The generator electromechanical dynamics are caused by the energy exchange between the mechanical power source and the electrical network in form of power demand. This interaction is commonly represented by the equation of motion showing the balance in the generator acceleration torque [118]. The electromagnetic dynamics represent the electrical current and magnetic flux interactions within the generator's internal windings. These interactions in the internal windings are represented based on Park's transformation, which simplifies the electrical dynamics using equivalent circuits in the rotor direct axis (d-axis) and quadrature axis (q-axis) [118].

The stability analysis in the present thesis focuses on the transient response of synchronous generators following a disturbance. The level of detail of the synchronous generator models varies depending on the dominant transient phenomenon and the number of electrical circuits considered to represent the rotor windings. The difference in time response of transients through generator windings results in classification of the generator representation into either the subtransient or transient electromotive force (emf) with the corresponding reactances [23]. In the present thesis, the generator is modeled using the transient emf. Two types of generator models with a different level of complexity and accuracy are applied: the fourth-order model and the second order (classical) model. The simplifying assumptions considered in the two models and the resulting differential and algebraic equations constituting the mathematical models of the generators are described in the following.

Fourth-Order Generator Model

The fourth-order model is represented by transient emfs in the d-axis, $E_d^{'}$, and in the q-axis, $E_q^{'}$, behind the respective transient reactances $X_q^{'}$ and $X_d^{'}$ of the generator. In this model, the generator construction is simplified by neglecting the effect of the d-axis damper windings [23].

The resulting structure of the model consists of an excitation field circuit on the d-axis and an equivalent damper winding circuit on the q-axis. This model is referred to in the literature as being accurate enough to represent synchronous generators for studying the electromechanical transient behavior [23], which is the scope of the current thesis. Thereby, the differential equations of this generator model comprise of Equations (2.1) and (2.2) describing the change in transient emfs due to decaying flux in the rotor circuit, and the equations representing the change in rotor angle and speed given by the equation of motion in Equations (2.3) and (2.4).

$$T_{d0}^{'}\dot{E}_{q}^{'} = E_f - E_{q}^{'} - (X_d - X_{d}^{'})I_d \tag{2.1}$$

$$T_{q0}^{'}\dot{E}_{d}^{'} = -E_{d}^{'} + (X_q - X_{q}^{'})I_q \tag{2.2}$$

$$\dot{\delta} = \omega - \omega_0 \tag{2.3}$$

$$\dot{\omega} = \frac{1}{M}\Big[P_m - D(\omega - \omega_0) - P_e\Big] \tag{2.4}$$

From Equations (2.1) and (2.2), E_f is the generator internal excitation voltage, $T_{d0}^{'}$ and $T_{q0}^{'}$ are the open-circuit d-axis and q-axis transient time constants, respectively, and I_d and I_q are the respective d- and q-axis components of the armature current. In Equations (2.3) and (2.4), M is the moment of inertia given by $M = \frac{2H}{\omega_0}$, where H is the inertia constant in seconds, P_m is the mechanical input power, P_e is the air gap electrical power, D is the damping coefficient, δ is the rotor angle, ω is the generator rotational speed, and ω_0 is the synchronous rotational speed.

According to the assumptions described in Section 2.1, the network voltages and currents are represented as fundamental phasor quantities. This results in simplification of the stator model by ignoring the stator transients. The stator model is therefore represented using an impedance consisting of the stator resistance R and the d- and q-axis components of the reactance. The algebraic equations representing the stator is given as

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} E_{d}^{'} \\ E_{q}^{'} \end{bmatrix} - \begin{bmatrix} R & -X_{q}^{'} \\ X_{d}^{'} & R \end{bmatrix} \begin{bmatrix} I_d \\ I_q \end{bmatrix} \tag{2.5}$$

where V_d and V_q are the d- and q-axis components of the generator terminal voltage. The set of algebraic equations of the generator is completed by the equation of the generator air gap power given by

$$P_e = (E_{d}^{'}I_d + E_{q}^{'}I_q) + (X_{d}^{'} - X_{q}^{'})I_d I_q. \tag{2.6}$$

The changes in emf E_f in Equation (2.1) and mechanical power P_m in Equation (2.4) are computed from the equations describing the excitation system model and the turbine-governor model, respectively, described in Section 2.2.2.

Second-Order Generator Model

The second-order generator model, also known as the classical model, is usually applied for simplified power system transient studies. It provides a reduced representation of the generator with no rotor windings on the d- and q-axis. The damper circuits and field flux decay are

neglected, and therefore ignores all state variables for the rotor coils. In formulation of the model equations, the changes in the internal excitation voltage E_f and the d-axis armature current I_d are assumed to be negligible during the transient state [23]. The generator is therefore represented by a constant emf $E^{'}$ behind a reactance $X_d^{'}$. The simplified generator model can be derived for the subtransient, transient and steady state conditions depending on the considered emf and the respective reactance. For the transient condition, which is of interest in the present thesis, the generator is represented using a constant transient emf behind transient reactance [119]. The system of differential equations describing the model is reduced to the equation of motion given in Equations (2.3) and (2.4). It is important to note that the input mechanical power P_m is assumed to be constant for studies applying the classical model. The algebraic equation representing the stator voltage variation is given by

$$V = E^{'} - jX_d^{'}I \tag{2.7}$$

where V is the generator terminal voltage with phase angle θ, and I is the armature current. The magnitude of the machine internal voltage is kept constant during transient analysis. However, the internal angle δ is variable and accounts for the rotational dynamics of the generator rotor [119]. The generator air gap power is computed by

$$P_e = \frac{E^{'}V}{X_d^{'}} \sin(\delta - \theta). \tag{2.8}$$

2.2.2 Synchronous Machine Control

The generator's dynamic response is affected by the action of the connected controllers. The controllers considered in the present thesis are the excitation system and the turbine-governor system. Excitation systems regulate the generator internal voltage through the supplied field current. Turbine-governor system regulates the mechanical input power by varying the inflow rate of the fluid driving the turbine. The current section describes the mathematical models of the controllers applied in the present thesis.

Excitation System

The excitation system model applied in the present thesis is a field-controlled dc commutator exciter with a continuously acting voltage regulator as described in [120]. The dynamics of the exciter are described by the following equations:

$$\dot{V}_r = \frac{1}{T_a}\Big[K_a(V_{ref} - V_f - V_c + V_s) - V_r\Big] \tag{2.9}$$

$$\dot{V}_f = \frac{1}{T_f}\left[\frac{K_a}{T_e}(V_r - V_x - K_e E_f) - V_f\right] \tag{2.10}$$

$$V_x = E_f A_{ex} \exp(B_{ex} E_f) \tag{2.11}$$

$$\dot{E}_f = \frac{1}{T_e}\Big[(V_r - K_e E_f - V_x)\Big] \tag{2.12}$$

In Equation (2.11), the exponential term $S_e = A_{ex} \exp(B_{ex} E_f)$ is the saturation function, where A_{ex} and B_{ex} are exponential function parameters. V_r is the regulator output voltage used for controlling the exciter, V_f is the stabilizing feedback voltage through a gain K_f with a time constant T_f. V_s is the power system stabilizing output signal, V_c is the measured generator terminal voltage and V_{ref} is a set reference voltage. K_a and T_a are the amplifying gain and time constant associated with the regulator. T_e is the integrating time (also known as the exciter time constant) and K_e is the negative feedback gain at the exciter stage. Additional details of the exciter model are given in [116].

Turbine–Governor System

The turbine-governor system applied in the computation methods developed in the present thesis is based on the TGOV1 steam turbine generation system (represented by a simple turbine model) and a droop controlled governor model as described in [121]. The control model for adjusting the turbine-valve position is based on a proportional-integral (PI) controller. The dynamics of the turbine-governor system are given by Equations (2.13) to (2.17).

$$y_1 = P_{ref} - K_R(\omega - \omega_{ref}) - P_m \tag{2.13}$$

$$\dot{y}_2 = \frac{1}{T_p} y_1 \tag{2.14}$$

$$y_3 = K_p y_1 + y_2 \tag{2.15}$$

$$\dot{y} = \frac{1}{T_m}(y_3 - y) \tag{2.16}$$

$$\dot{P}_m = \frac{1}{T_k}(y - P_m) \tag{2.17}$$

Variable y is the control valve position, P_{ref} is the reference power, P_m is the mechanical power output of the turbine, ω_{ref} is reference speed and ω is the rotor speed. The constant $K_R = \frac{1}{R}$, where R is the droop constant, which determines the required change in power with respect to the frequency deviation [96]. The variable y_1 is the input power deviation to the controller. K_p and T_p are the gain and time constant of the PI controller, respectively. Variables y_2 and y_3 are intermediate states of the PI controller. T_m and T_k are the time constants of the valve position controller and turbine, respectively.

The above turbine-governor model is represented using a simple steam turbine. A more detailed turbine-governor system is described in [116, 122] using actual thermodynamic characteristics to represent the steam turbine stage. For the purpose of comparison, the detailed model is implemented and tested in an open source object oriented phasor-time-domain simulation toolbox, OpenModelica [38], which is based on the Modelica modeling language [123]. Detailed results are presented in [116, 122].

2.2.3 Network Model

Network components are accurately modeled using wave equations, which define the relation of network voltages and currents depending on time and position as applied in the detailed electromagnetic transient analysis [24]. However, as described in Section 2.1, transient stability studies assume that the changes in the propagating voltage and current waves are very fast compared to the resulting dynamics in the rotating machines. Thereby, the network is assumed to be in a quasi-steady state, changing from one equilibrium operating point to another. With this assumption, the transmission line model in transient stability analysis is similar to the model applied in steady state stability studies. The transmission line is therefore modeled using lumped-parameters as a pi-equivalent circuit with a series impedance Z_s and a shunt admittance Y_{sh} as described in [116]. Formulation of the network equation in [116] using the nodal admittance matrix shows that network elements are symmetrical.

Another component considered as part of the network subsystem model is the transformer. In a simplified form, a network transformer is represented by an ideal transformer with a complex transformation ratio in series with an impedance [116]. For the scope of analysis in the present thesis, the transformers are considered to have fixed voltage ratios and fixed phase shifts. From the simplified transformer circuit, a nodal admittance matrix formulation is derived for the representation of the transformer as described in [116].

The resulting transmission line and transformer model representation are shown to have a similar matrix structure. For the steady state and transient stability analyses considered in the present thesis, the transmission line and transformer elements are combined in the formulation of the network model and are referred to as branch elements. The main model parameters of the branch elements are the resistance, inductance and shunt capacitance, with which the resulting nodal admittance matrix Y is formulated. The individual line models are combined to form the whole network based on Kirchhoff's current law. The nodal network equation in the complex current balance form is given by

$$I = YV \tag{2.18}$$

where;
Y is the nodal admittance matrix
V is a vector of phasor node voltages
I is a vector of phasor current injections into the network nodes.

The nodal admittance matrix is formulated from the individual branch elements as follows:

- The diagonal elements of the Y matrix are the self-admittances of the corresponding nodes. These values are computed as the sum of all the admittances of the branches connected to the corresponding node.

- The off-diagonal elements of the Y matrix are the mutual admittances between two interconnected nodes. The admittance is equal to the negative of the series admittance of the branch connecting the corresponding nodes. An off-diagonal element of the admittance matrix is zero if there is no branch linking the two nodes [23]. In a typical power system network, there are few interconnected nodes, which means many zero off-diagonal

elements of the matrix. The resulting admittance matrix therefore has a high sparsity percentage that increases with the network size.

2.2.4 Load Modeling

A load in power system studies may represent an individual consumption device, an aggregation of individual loads at a specific node or an implicit representation of a network subsection in the system model. The simplification applied in modeling large networks in transient stability analysis represents network subsections, especially the distribution networks, in form of aggregated loads at the interconnecting nodes. The net effect of the aggregated loads can be represented as static or dynamic load models. Static loads model the load characteristics at a given instant as a linear or nonlinear function of the node voltage and frequency using algebraic equations. Although this modeling does not represent the actual behavior of the loads, it is acceptable under the assumption that the loads have a very fast response to reach steady state operation compared to the transient phenomena in the rotating machines under consideration. However, for studies where the dynamic behavior of loads plays a significant role, dynamic load models are used. Dynamic loads are represented using differential equations. The scope of the load models used in the present thesis is limited to static loads with voltage dependency.

The general representation of the voltage dependency of the static load characteristic is given by the exponential model

$$P = P_0 \left(\frac{V}{V_0} \right)^a \tag{2.19}$$

$$Q = Q_0 \left(\frac{V}{V_0} \right)^b \tag{2.20}$$

where V_0 is the initial voltage at the load bus obtained from steady state calculations, P_0 and Q_0 are the corresponding active and reactive power values at the initial operating conditions, and P and Q are the active and reactive power components of the load at voltage V. The active power exponent a, and the reactive power exponent b are the parameters of the model. For $a = b = 0, 1, 2$, the model represents constant power, constant current or constant impedance characteristics, respectively [96, 124].

Unlike in the steady state power flow analysis, where the loads are represented as constant power loads (i.e. $a = b = 0$), loads applied for transient stability analysis in the current thesis are represented as constant impedance loads (i.e. $a = b = 2$). This supposition is taken as part of the system complexity reduction measures; constant impedance load models eliminate the current injections at the load buses, which results in a linear representation of the loads where the load admittance is calculated using Equation (2.21). With this simplification, the load admittance is directly added to the nodal admittance matrix given in Equation (2.18) at the respective diagonal element.

$$Y_l = \frac{P_0 - jQ_0}{|V_0|^2} \tag{2.21}$$

The assumption of a quasi-steady state network and the constant impedance static load models result in the overall network subsystem being represented by a linear model formulated in terms of an admittance matrix. This reduces the model of the network subsystem to a linear system of equations.

2.2.5 Generator – Network Interface

In addition to the model formulation of the generator subsystem and network subsystem components, it is necessary to describe how the two subsystems are interfaced to form the complete power system model. An important factor which is not evident in the described models is the difference in the reference coordinates considered during the modeling of the two subsystems. The two reference frames are the generator's d-q orthogonal reference frame rotating with the rotor (represented with coordinates d-q) and the stationary network's complex coordinate reference frame (represented with coordinates D-Q). Figure 2.1 depicts the relationship between the two reference frames.

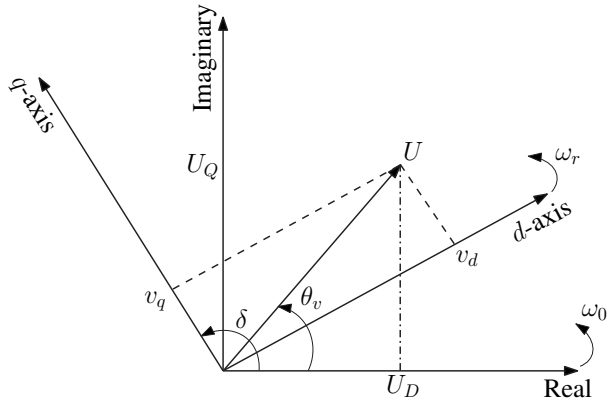


Figure 2.1: Reference frame transformation

The equations describing the behavior of the generator are formulated in the rotor d-q orthogonal reference coordinates. This representation simplifies system modeling and solution of the generator equations by transforming the time-varying stator quantities from the stationary network coordinate reference frame to a reference frame which rotates with the rotor [96]. The resulting generator equations need to be linked to the network equations in order to form a systematic interconnection of the generator subsystem to the rest of the network. However, the network equations are represented in the network's complex coordinate reference frame. A system transformation technique is therefore required to relate the quantities in the two reference systems. The transformation and inverse transformation between generator quantities (v_d and v_q) and the network quantities (U_D and U_Q) derived from the phasor diagram in Figure 2.1 are given by Equations (2.22) and (2.23) [96].

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \sin\delta & -\cos\delta \\ \cos\delta & \sin\delta \end{bmatrix} \begin{bmatrix} U_D \\ U_Q \end{bmatrix} \tag{2.22}$$

$$\begin{bmatrix} U_D \\ U_Q \end{bmatrix} = \begin{bmatrix} \sin\delta & \cos\delta \\ -\cos\delta & \sin\delta \end{bmatrix} \begin{bmatrix} v_d \\ v_q \end{bmatrix} \tag{2.23}$$

In Equations (2.22) and (2.23), the variable δ is the rotor angle, which defines the position of the rotor relative to the stator reference axis. In this representation, the rotor angle δ is the displacement between the generator q-axis and the real axis (D-axis) of the network. The generator q-axis is used as the reference axis of the rotor due to the fact that the nominal induced voltage vector, which represents the rotor position, lies on the q-axis. It is important to note that the number of d-q system coordinates is equal to the number of generators in the network since each generator may operate at a different rotor angle [23]. Moreover, the transformations in Equations (2.22) and (2.23) are applicable to all generator and network variables.

2.3 Modified Models for Asymmetrical Transients' Analysis

In the previous section, the models described for the basic transient stability analysis are limited to balanced three-phase operating conditions. One of the contributions in the current thesis is to extend system analysis to asymmetrical network transients. For this, appropriate component models are required for the asymmetrical transients' analysis computational method. The current section describes the mathematical modeling of the fundamental power system components considering balanced and unbalanced operating conditions. The modeling is based on the symmetrical components method that represents the system model using three sequence networks.

2.3.1 Modeling Basis

The modeling of the components for the asymmetrical transients' stability analysis domain in the present thesis is based on the quasi-stationary symmetrical components method. Generally, power system variables are represented as a set of three-phase time varying phasors corresponding to the three-phase system. The symmetrical components method is used to resolve a set of three unbalanced phasors in the three-phase system into three symmetrical components; positive-, negative- and zero- sequence components. This is achieved using the symmetrical components transformation matrix (T_{sym}) in Equation (2.24) [125] to transform the phasor quantities into sequence quantities.

$$V^{abc} = T_{sym} V^{012} \tag{2.24}$$

where $T_{sym} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & a^2 & a \\ 1 & a & a^2 \end{bmatrix}$ and $a = 1\angle 120^o$.

$V^{abc} = \begin{bmatrix} V^a & V^b & V^c \end{bmatrix}^T$ is a vector of the respective phasor voltages in phases A, B, C, and $V^{012} = \begin{bmatrix} V^0 & V^1 & V^2 \end{bmatrix}^T$ is a vector of the sequence voltages in the zero, positive and negative sequence circuits. This relation also applies for the transformation between phasor and sequence currents.

The simplifying assumptions considered during component modeling are described in Section 2.1. Unlike the modeling in Section 2.2, the modeling described in the current section considers both balanced and unbalanced network operating conditions. The network is represented using the positive, negative, and zero sequence networks. The sequence circuits are decoupled during normal operation. This implies that a given sequence current will only affect the voltage drop in that sequence circuit, and the voltage remains unchanged in the other circuits. This property allows the entire system including generators, transmission lines and transformers to be represented by three decoupled sequence systems [125] during normal operation.

2.3.2 New Synchronous Generator Model

Based on the symmetrical components method, the synchronous generator dynamics are divided into the positive, negative and zero sequence components, which form a set of three decoupled circuits. Assuming machine physical symmetry constraints, whereby the generator windings are designed to be balanced, the resulting generator internal voltage is a balanced three-phase voltage [45]. This implies that there are no negative and zero sequence current sources during normal operating conditions. The voltages and currents in the negative and zero sequence circuits result from unbalanced network operations. Therefore, the negative and zero circuit power generation is set to zero. This property is the basis of the new synchronous generator model defined in the current section. The mathematical description of the machine model in the individual components is explained in the following.

Generator Positive Sequence Dynamics

The generator positive sequence circuit is represented using the balanced transient stability synchronous generator models. This is an acceptable representation since the models used in balanced transient stability studies are developed considering only the positive sequence operation of the generator. Therefore, the differential and algebraic equations describing the dynamics of the generator positive sequence network are similar to the equations derived in Section 2.2.1 for the balanced system. However, the computed voltages and currents in this case only refer to the positive sequence circuit of the machine, unlike in the balanced simulation mode where the quantities refer to the total machine quantities.

Generator Negative Sequence Dynamics

Due to the generator's physical symmetry constraints, there is no current source in the negative sequence circuit during normal operation. As a result, the equivalent generator negative circuit is represented by a pure impedance connected between the synchronous generator bus and ground. The relationship between the negative sequence stator voltage and current is given by

$$0 = V_2 + Z_2 I_2 \tag{2.25}$$

where;
I_2 is the complex negative sequence current,
V_2 is the complex negative sequence voltage at the generator bus, and

Z_2 is the negative sequence impedance given by $Z_2 = R_s + jX_2$. R_s and X_2 are the stator resistance and negative sequence reactance, respectively.

Generator Zero Sequence Dynamics

Similar to the negative sequence circuit, there is no current source in the zero sequence circuit during normal operation. The zero sequence currents result from network imbalances. Therefore, the equivalent generator zero sequence circuit is also represented by a pure impedance connected between the generator bus and ground. Equation (2.26) gives the relationship between the zero sequence stator voltage and current.

$$0 = V_0 + Z_0 I_0 \tag{2.26}$$

where;
I_0 is the complex zero sequence current,
V_0 is the complex zero sequence voltage at the generator bus, and
Z_0 is the zero sequence impedance given by $Z_0 = R_s + jX_0$, where X_0 is the zero sequence reactance.

As described in [54], the zero sequence stator components vanish in the rotor d-q reference frame. The alternative representation as described in [54] is to express these dynamics in the real and imaginary coordinates as

$$\begin{aligned} v_0(t) &= V_0 \cos(\omega_s t + \theta_0) \\ v_0(t) &= V_R(t)\cos(\omega_s t) - V_I(t)\sin(\omega_s t) \end{aligned} \tag{2.27}$$

whereby V_0 is the magnitude of the zero sequence quantity with the corresponding phase angle θ_0, and ω_s is the synchronous rotation speed. The variables V_R and V_I are the real and imaginary components of the phasor representing the zero sequence stator quantity, respectively [54], given by $V_R(t) = V_0 \cos\theta_0$ and $V_I(t) = V_0 \sin\theta_0$. The expressions in Equation (2.27) apply for both voltages and currents in the stator.

Generator Mechanical Equations

The mechanical representation of the generator for asymmetrical transients is derived by modifying the mechanical equations used in the symmetrical transient stability analysis models, given in Equations (2.3) and (2.4), to include the effect of unbalanced system operation. It should be noted that the positive sequence current represents the main electrical torque in the system. An additional term representing a braking torque is included in the mechanical equation of the generator to account for the resulting negative sequence currents that produce an opposing torque. The zero sequence currents do not produce an effective torque in the machine and are not included in the mechanical equation. As a result, the mechanical dynamics of the synchronous machine are given by the modified swing equation as

$$\dot{\delta} = \omega - \omega_0 \tag{2.28}$$

$$\dot{\omega} = \frac{1}{M}\Big[P_m - D(\omega - \omega_0) - P_e - (R_2 - R_s)I_2^2\Big] \tag{2.29}$$

where R_2 is the negative sequence resistance [55]. In Equation (2.29), the electrical power represents the generator power from the three sequence circuits. However, since the negative and zero sequence circuits have no voltage sources, the resultant air gap electrical power of the generator is only due to the positive sequence component. Therefore, the air gap power is given by Equation (2.6).

Machine Controllers

As described in the preceding section, the negative and zero sequence networks have no voltage sources. The controllable generator field voltage is therefore located in the positive sequence network, similar to the model in the balanced system analysis. Consequently, the model of the generator excitation control system applied during analysis of asymmetrical transients is similar to the model described by Equations (2.9) – (2.12) in Section 2.2.2. Similarly, since the electrical power output of the generator only results from the positive sequence component, the turbine-governor system applied for studying asymmetrical transients is also similar to the model used for the balanced system analysis described by Equations (2.13) – (2.17).

2.3.3 Network Modeling

The network is modeled using the quasi-steady state assumption as applied to the symmetrical transients' analysis. However, the three-phase network in this case is represented by three decoupled single-line sequence circuits for the positive, negative, and zero sequence networks. The sequence networks are constructed using equivalent admittances. The relation between the injected sequence currents at the nodes and the sequence node voltages is given in the complex current balance form as

$$\begin{aligned} I_0 &= Y_0 V_0 \\ I_1 &= Y_1 V_1 \\ I_2 &= Y_2 V_2 \end{aligned} \tag{2.30}$$

where;
Y is the nodal admittance matrix for the corresponding sequence network,
V is a vector of sequence node voltages, and
I is a vector of injected sequence currents into the nodes. In Equation (2.30), $0, 1, 2$ refer to the zero, positive and negative sequence of the respective quantities. In the formulation of the network equations, the network is assumed to operate at a constant frequency. The voltages and currents are represented as phasors at fundamental frequency.

The generator stator and loads are modeled as part of the network. In this case, the nodal admittance matrices of the sequence networks are augmented by adding the generator stator admittance and load admittance to the diagonal shunt elements at the respective network nodes to which generators and loads are connected.

Generator Representation

The generator stator sequence admittances (Y_{g0}, Y_{g1}, Y_{g2}) are given by

$$\begin{aligned} Y_{g0} &= 1/Z_0 \\ Y_{g1} &= 1/Z_1 \\ Y_{g2} &= 1/Z_2 \end{aligned} \tag{2.31}$$

where Z_0, Z_1, Z_2 are the zero, positive and negative sequence impedances of the stator.

Load Representation

Static loads are modeled as constant admittances for the asymmetrical transient studies, similar to the assumption in the balanced transients' studies, in order to simplify the system modeling by eliminating current injections from the loads. However, since the network model is separated into sequence networks, the loads are also modeled in terms of their sequence component admittances. The elements of the load admittance diagonal matrix – in terms of the phasor representation Y^i_{abc} at the i^{th} node – are derived from the initial load power $P^i_{abc}+jQ^i_{abc}$ and the load bus voltage V^i_{abc} in the phase coordinates using Equation (2.32). The load admittance Y^i_{012} in terms of sequence components is calculated using the symmetrical components transformation matrix using Equation (2.33) [125].

$$Y^i_{abc} = \frac{P^i_{abc} - jQ^i_{abc}}{|V^i_{abc}|^2} \tag{2.32}$$

$$Y^i_{012} = [T_{sym}]^{-1}[Y^i_{abc}][T_{sym}] \tag{2.33}$$

Considering the combined network subsystem, the three sequences are independent during balanced system operating conditions. Therefore, the sequence components are modelled and analyzed separately on a per phase basis. The phase quantities (currents and voltages) are computed by superimposing the sequence component quantities. For an unbalanced fault, the sequence admittance matrices are initially computed separately and the networks connected together using the asymmetrical fault analysis techniques depending on the fault type as described in [125]. System variables during the fault period are determined from the solution using the equivalent network connections.

2.3.4 Reference Frame Transformation

The reference frame transformation is applied to interface the generator and the network subsystems. As described in Section 2.2.5, the equations of the generator subsystem are developed in the rotor d-q reference frame, while the network subsystem is modeled in the complex coordinate reference frame. In order to account for the sequence components, the complex coordinate reference frame in this case is divided into the D_1-Q_1 and D_2-Q_2 sequence network reference frames; where the D_1-Q_1 axis is similar to the coordinate reference frame described in Section 2.2.5 rotating synchronously in the positive sequence direction, and the D_2-Q_2 axis represents the network reference frame rotating synchronously in the negative sequence direction. The

relative position of the d-q reference frame and sequence network complex coordinates is illustrated in Figure 2.2. The d-q axis represents the axis fixed on the generator field magnetic axis of the respective machine and rotates with the rotor.

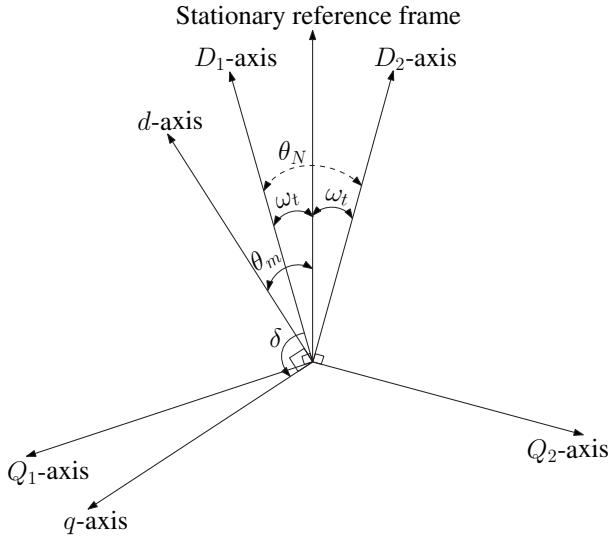


Figure 2.2: Positive and negative sequence components reference frame transformation

The different angular displacements in Figure 2.2 are defined as follows: The angle between the real axis of the positive sequence reference D_1-axis and the stationary frame is defined by ωt, where ω is the synchronous angular velocity; Angle θ_m between the d-axis and the stationary reference frame represents the angular position of the rotor; δ is the machine rotor angle as described in Section 2.2.5 [47]. From Figure 2.2, the following transformations between machine quantities (v_d,v_q) and the respective sequence network quantities (U_D, U_Q) are derived:

$$\begin{bmatrix} v_{d1} \\ v_{q1} \end{bmatrix} = \begin{bmatrix} \sin\delta & -\cos\delta \\ \cos\delta & \sin\delta \end{bmatrix} \begin{bmatrix} U_{D1} \\ U_{Q1} \end{bmatrix} \tag{2.34}$$

$$\begin{bmatrix} v_{d2} \\ v_{q2} \end{bmatrix} = \begin{bmatrix} \cos\theta_N & -\sin\theta_N \\ -\sin\theta_N & -\cos\theta_N \end{bmatrix} \begin{bmatrix} U_{D2} \\ U_{Q2} \end{bmatrix} \tag{2.35}$$

where $\theta_N = 2\omega t + \delta - \pi/2$ is the displacement angle between the D_2-axis and the d-axis, and ω is the synchronous rotational speed in the negative sequence direction [47]. The relations in Equations (2.34) and (2.35) hold for the transformation of currents and voltages. The zero sequence transformation is not considered here since the zero sequence stator quantities are nonexistent in the d-q coordinates.

2.4 Integrated Power System Model

Having illustrated the modeling of the fundamental components of the power system in the previous sections, the current section briefly describes the general form of the power system

model. Figure 2.3 shows the interconnection between the main components of a power system for transient stability analysis and the key variables exchanged between the components. The full power system model for transient stability studies is described by a set of differential and algebraic equations of the form

$$\dot{x} = f(x, y, u) \tag{2.36}$$

$$0 = g(x, y, u) \tag{2.37}$$

where x is a vector of dynamic state variables associated with the differential equations set, y is a vector of algebraic variables, and u are system parameters.

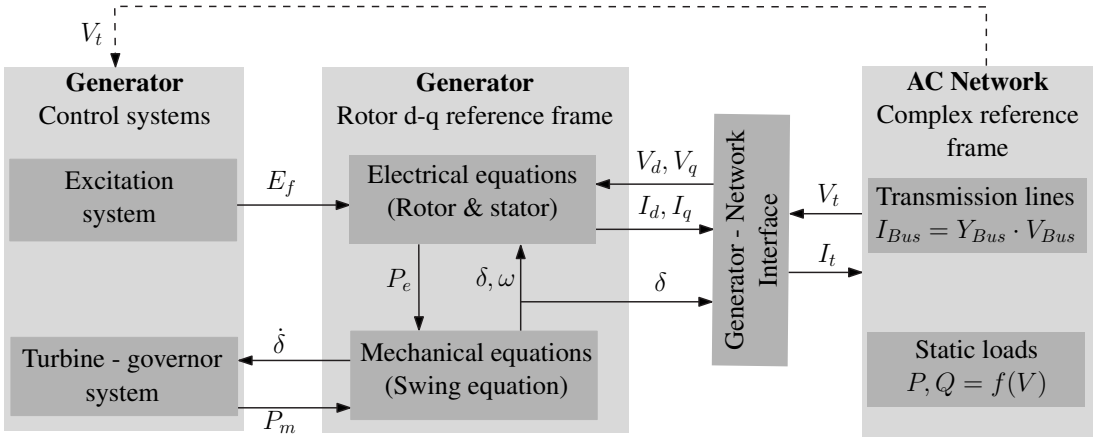


Figure 2.3: Interconnection of power system components

The differential equation is a set of uncoupled subsets representing all synchronous machines in the system. This property is important for the parallelization of the solution of differential equations proposed in the current thesis as described in Chapter 5. The equations of the synchronous machines are divided into the rotor electrical and mechanical equations, and control system equations. The different generator subsystems are coupled to each other through the network. Other components that are represented by differential equations but not included in the modeling presented in the current thesis include Static Var Compensators (SVC), dynamic loads, HVDC and FACTS devices. The set of algebraic equations comprises the stator equations of each machine expressed in the network frame of reference, coupled to the equations of the transmission network and static loads. The interface between the synchronous machines and network system is through a set of transformations represented by algebraic equations as described in Section 2.2.5 and Section 2.3.4.

The power system model normally uses a single line representation for balanced transients' studies. A similar generic structure is used for asymmetrical transients' studies, but with the network represented using the three sequence circuits connected to the generator system via the transformations derived in Section 2.3.4. The three circuits are decoupled during normal operation and interconnected during fault period based on the asymmetrical fault analysis techniques.

2.5 Summary

The current chapter addresses part of the mathematical modeling objective of the present thesis. As an outcome of the chapter is the modeling of the conventional power system components applied for transient stability analysis under balanced network operating conditions – during normal and fault conditions – and unbalanced network operations as a result of asymmetrical network transients. The system models are described by a set of differential and algebraic equations representing the dynamic behavior of individual components based on quasi stationary phasors. The quasi stationary phasor approximation considered during modeling is acceptable since the analyzed transients are slow compared to the propagating signals. Since the main source of the analyzed electromechanical transients is the generator, model details of the generator and the controllers connected to the generator are important regarding system representation and analysis accuracy. In general, the assumptions considered during system modeling are necessary to simplify the overall system model and reduce computational complexity in the analysis methods described in the next chapter.

3 Time-Domain Simulation of Conventional Power System

System stability analysis involves the solution of the equations resulting from the system modeling with a general form given by Equations (2.36) and (2.37). The power flow analysis approach is the commonly used method to derive the system behavior considering steady state operating conditions. To determine the time response of the system, transient stability analysis is applied. Time domain simulations are the basis for transient stability analysis in the existing software packages applied in research and industry. The system solution is derived via step-by-step numerical integration of the differential algebraic equations describing the power system. However, the solution method used in most simulation tools is derived mainly assuming symmetrical networks and operating conditions, which might no longer be accurate in the current power system operating environment.

The present chapter describes the development of a Matlab-based time-domain computational method for the analysis of balanced and unbalanced power system transients. In addition to the conventional transient stability analysis method, where the transmission network representation is based on a per phase positive sequence, an asymmetrical simulation algorithm is developed to form the combined computational method. The asymmetrical simulation algorithm is based on the symmetrical components technique in which the network is represented by three sequence circuits. The method is developed as an extension to an analysis approach in a research-based toolbox, MatDyn, for studying balanced transients. An overview of MatDyn toolbox is described in Section 3.1 as a basis for the functional modifications in the extended computational method. Implementation details and the new solution methodology in the extended analysis method are presented in Section 3.2. Furthermore, Section 3.3 presents the simulation results obtained using the extended method and validation against the commercial grade software package DIgSILENT PowerFactory in terms of analysis accuracy.

3.1 Symmetrical Transient Stability Analysis Method

Before presenting the extension of the computational method to asymmetrical power system analysis, it is of interest to describe the fundamental building block of the developed compu-

The details described in the current chapter are based on the publications [117, 126].

M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Dynamic Simulation Module for Power System Transients Analysis in the *e*ASiMOV Framework," in *2017 European Modelling Symposium (EMS)*, Manchester, UK, pp. 157–162, November 2017.

M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Simulation Tool for the Analysis of Unsymmetrical Power System Transients in Large Networks," in *European Conference on Modelling and Simulation (ECMS)*, Wilhelmshaven, Germany, pp. 246–253, May 2018.

tational algorithm. The computational method is based on the symmetrical transient stability approach as applied in conventional transient simulation software packages for balanced system analyses. As a first step of the development, the current section presents an overview of the transient simulation method providing the basis of the computational method extended to analysis of symmetrical and asymmetrical power system transients. The main components of the basic algorithm are also presented to illustrate the interaction between the different stages of the computation, which provides insight into the necessary modifications in the system structure as well as the numerical solution technique for the new method.

3.1.1 Overview of the Symmetrical Transient Analysis Method

The symmetrical transient stability analysis (TS) function is the basis in the computational method for the time-domain simulations presented in the current thesis. In this method, the analysis is based on symmetrical and steady-state representation of the network and therefore ignores the fast transients in transmission lines. For this reason, the assumptions stated in Section 2.1 and the system models derived in Section 2.2 apply for the computational simulations under this category. The symmetrical transient stability method is based on MatDyn [40], which is an open source simulation toolbox in Matlab for analysis of power system transients under balanced network conditions. The development of the MatDyn software toolbox is based on Matpower [33], a Matlab toolbox for power flow and optimal power flow computations. With the analysis limitation of the Matpower toolbox to steady state computations, MatDyn extends the analysis domain to include transient stability analysis and time domain simulation of power systems.

An important feature in the MatDyn toolbox is that the transient analysis method directly applies the optimized and efficient functions in the Matpower library for the relevant steady state precomputations at the initial stage of the simulation. In addition, the toolbox provides a flexible modeling environment, which is important for model modification and extension of its functionality to include additional analysis features. This property provides the basis for the extension of the time-domain computational method to include the analysis of asymmetrical network transients in system stability studies as described in Section 3.2.

3.1.2 Components of the Symmetrical Transient Analysis Method

The structure of the computational method for symmetrical transient stability analysis can be viewed in terms of the model composition and numerical solutions. Component models in the system are divided into steady state models grouped under the network subsystem represented by algebraic equations, and dynamic models mainly represented by differential equations. The solution methodology is divided in such a way that the solution of the steady state system equations is done using power flow computations, while the dynamic system equations are solved using numerical integration and algebraic solvers over time. The components in the basic transient stability toolbox are defined as a collection of Matlab m-files. Figure 3.1 depicts an overview of the main code blocks in the toolbox as well as the interaction between the different functions in the computation process.

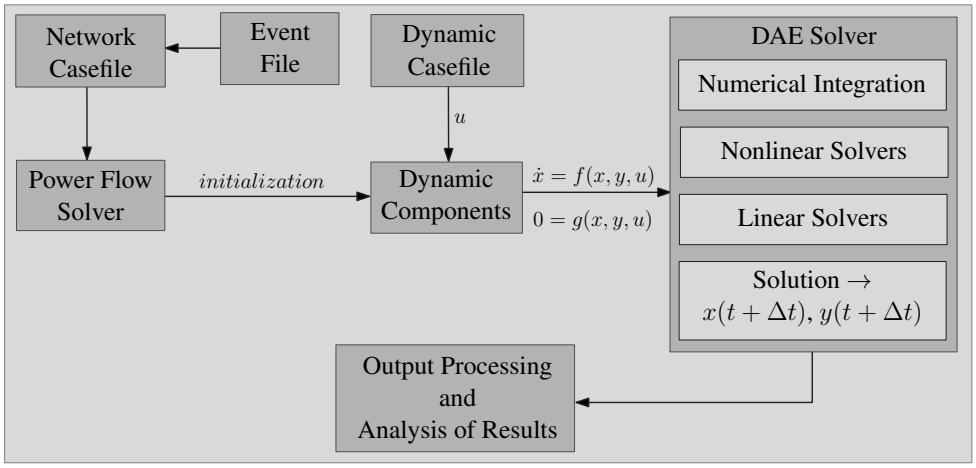


Figure 3.1: Block diagram of the dynamic simulation module

The computational analysis starts with the input of the structure of the network to be analyzed. From the structural representation, a network file is defined in text format using bus indices and the corresponding transmission line parameters connecting the buses in terms of resistance, inductance and capacitance. As described in Chapter 2, the network representation in dynamic simulations is similar to that used in steady state analysis. Therefore, the network structures defined in the steady state simulation toolbox, Matpower, are directly applied as input files to the dynamic computational simulation method. The input network structure is defined in a Matlab file referred to as the *Network Casefile* in Figure 3.1. The following blocks constitute the network casefile: Bus data block – defining the components connected to the specific buses; Generator data block – defining the rating and location of the machines connected to the network; and Branch data block – specifying the parameters of the network branches.

After defining the input network structure, the nodal admittance matrix (Y_{Bus}) is formulated using a Matlab function from the Matpower library. The Y_{Bus} matrix is used in the network equation given in current balance form in Equation (2.18) for the computation of the bus voltages using defined node current injections. For application to dynamic simulations, the nodal admittance matrix is modified to include the coupling of the network to the generators and constant impedance loads. This interconnection is achieved by inserting the generator stator admittances and load admittances as diagonal shunt elements in the nodal admittance matrix.

As described in Chapter 2, the loads considered in the present thesis are static loads. These loads are defined as constant power loads in the network casefile. This format of load definition is directly applied for the steady state power flow simulations. For use in the dynamics simulation, however, the constant power loads are converted to a form that represents constant admittances. The load admittance Y_{load} at bus i is derived from the initial load power $P_i + jQ_i$ and the initial load bus voltage V_i as

$$Y_{load} = \frac{P_i - jQ_i}{|V_i|^2}. \tag{3.1}$$

Similarly, the admittance of the generator stator is added to the nodal admittance matrix to form the interconnection between the generation system and transmission network. In the symmetrical transient stability function, the generator stator is represented on a single phase basis since the same assumption is made for the network model. The equivalent generator stator admittance at a specific bus is given by

$$Y_{gen} = \frac{1}{R_a + jX_d^{'}} \tag{3.2}$$

where R_a is the stator resistance and $X_d^{'}$ is the d-axis transient reactance. The equivalent nodal admittance matrix consisting of the line parameters, generator stator admittance and constant impedance loads constitutes the algebraic equation of the network subsystem.

The first step of the computation process is the initialization stage. This is performed to compute the initial state of the algebraic and dynamic variables in order to progress the dynamic simulation from steady state. The steady state power flow computations are used to calculate the initial steady state bus variables, which include bus voltages – in terms of magnitude and angle – and the bus active and reactive power. This operation is represented in Figure 3.1 by the *Power Flow Solver* block. The power flow calculation is based on standard Newton's solver in the Matpower package [33].

During the time-domain simulation, system disturbances are considered to be predefined time based events. The *event file* block defines the required changes in the network structure during the simulation to represent the system disturbances. The file includes the event type and the time of event occurrence. The simulated event types include changes in bus parameters to simulate system faults, changes in line parameters to simulate line switching events, and variations in load parameters. A detailed description of the definition of the event types studied in the present thesis is given in [127]. In order to account for changes in network structure due to an event, the admittance matrix is reformulated at every occurrence of an event during the simulation.

The *dynamic components* block represents the mathematical description of the dynamic models in form of differential and algebraic equations. The block is a collection of Matlab m-files representing the different dynamic components in the system. Initialization functions are defined for each component to evaluate the steady state characteristics of the dynamic models at the beginning of the simulation. In the current chapter, the modeled dynamic components are the generator, exciter and the turbine-governor as described in Chapter 2. In addition to the mathematical description of the dynamic models, the parameters of the dynamic components are defined in the *dynamic casefile* function. The parameters in the dynamic casefile function depend on the network structure under consideration and the level of detail of the dynamic components.

With the network and dynamic models formulated, the DAE solver block represents the numerical solution of the resulting differential equations and algebraic equations. The solution consists of two main parts; the numerical integration of the differential equations and linear solution of the algebraic equations. MatDyn, which forms the basis of the symmetrical transient stability analysis function, consists of a number of integration solvers. The integration method for discretization of the differential equations in the current thesis is the fixed step size fourth

order Runge-Kutta method [128]. The solution of the algebraic equations is based on the direct linear solver using LU factorization algorithm from the Matlab library. Further details of the solution approach for the differential and algebraic equations, and integration solvers included in the toolbox are given in [40] and [127].

Algorithm 1 shows the pseudo code for the workflow of the dynamic simulation program. The *handle events* phase in the simulation workflow consists of reformulation of the combined node admittance matrix to account for the change in network topology and solution of the algebraic equations to compute the new state of algebraic variables. This process is carried out at every event of topology change before the next simulation time step.

Algorithm 1 : Overview of the workflow in the symmetrical transient stability analysis method

Inputs: Network structure data
Load event file
Load dynamic casefile
Compute power flow
Formulate augmented Y_{Bus} matrix
Initialize system to steady state $0 = g(x_0, y_0)$ and $0 = f(x_0, y_0)$
while $t < simulation\ time$ **do**
    Solve DAE: $\dot{x} = f(x, y)$ and $0 = g(x, y)$
    if *event happened* **then**
        Reformulate Y_{bus}
        Solve $0 = g_{new}(x_{old}, y_{new})$
    Update: x and y
    Save x_t and y_t variables at time t
    $t \leftarrow t + dt$
    Repeat: Until set simulation time is reached
Return: System variables x and y

The symmetrical transient stability analysis function described in current section deals with the analysis of power system performance following a disturbance under balanced network conditions. The studied events include three-phase short circuit faults, balanced load changes, changes in line parameters, and loss of generation. During the simulation, specific variables are saved at every time step. The simulation is advanced with a fixed step size. System stability is analyzed from the simulation results at the end of the simulation in form of Matlab plots. The main variables analyzed in this simulation function include generator power, rotational speed, rotor angle, and bus voltage for a specific period after the disturbance is cleared. A detailed analysis of the simulation results obtained using the symmetrical transient stability analysis function and a comparison to the results obtained using an open-source tool, OpenModelica, and a commercial grade software package, DIgSILENT PowerFactory, are given in [126].

3.2 Proposed Three-Phase Transient Stability Analysis Method

The symmetrical transient stability analysis function described in the previous section is limited to balanced networks and event types since the network model is based on the symmetrical representation. The current section describes the extension of the computational method to include the analysis of asymmetrical transients. Required modifications for asymmetrical transients' analysis and the solution methodology in the proposed simulation function are described in the following.

3.2.1 Overview of the Proposed Method

The method developed to include the analysis of asymmetrical network transients is referred to as the three-phase transient stability analysis method (3Ph-TS) in the present thesis. The three-phase transient stability function is developed to eliminate the limiting assumption of balanced transients by considering the analysis based on three sequence circuits. In this way, unbalanced network transients are analyzed by decomposing the analysis into the positive, negative, and zero sequence components. Using the sequence transformation matrix, further information can be derived from the sequence values about the individual phases during normal and fault operation. In other words, the analysis provides phasor current and voltage information for the positive, negative and zero sequences and for the A-, B-, and C-phases.

In comparison to the symmetrical transient function, the main difference regarding system modeling is in the algebraic representation of the network subsystem models. This also includes the generator stator and static load models, which are connected to the transmission network. The modifications in the component modeling and the simplifying assumptions applied for the extended computational method are described in Section 2.3.

The structural overview of the three-phase transient stability analysis function is also represented by the block diagram shown in Figure 3.1. The extended computational method is developed based on the numerical formulations of the symmetrical transient stability analysis function presented in Section 3.1. During the formulation of the extended analysis function, it was cautiously considered to maintain the same level of simplicity of the computational algorithm as the basic simulation method in order to allow for future functional improvements and scalability of the combined computational methods. The features of the extended simulation function are power flow analysis, dynamic analysis under balanced and unbalanced network operating conditions, and analysis of symmetrical and asymmetrical faults. In the following subsections, the modifications in the functional blocks and solution methodology to achieve the above mentioned features in the extended computational method are described.

3.2.2 Functional Modifications in the Proposed Method

The functions developed in the three-phase transient stability analysis method are extensions of the different functional blocks described in Section 3.1.2. The respective functions are modified to account for the changes introduced by the modeling for the unbalanced network behavior

and asymmetrical transients' analysis. Like in the symmetrical transient stability function, the network structure is formulated based on the Matpower casefile format. It is important to note that the basic Matpower casefile is a positive sequence network based definition. In order to account for the other sequence networks, the *Network Casefile* block is modified to include the bus and branch data for the negative and zero sequence circuits. This is equivalent to the construction of three independent network structures. The assumption of physical symmetry of the generator construction implies that there is no generation from the negative and zero sequence networks. Thereby, the generator steady state data in the casefile, which include power generation and internal generated voltage, are defined using only the positive sequence data. The bus and branch data of the negative sequence network are identical to the data defined for the positive sequence network. The zero sequence branch data are different from the negative and positive data since the zero sequence network topology differs from the positive sequence network topology [125].

During the computation process, it is assumed that the network is balanced, symmetrical and in steady state at the beginning of the simulation. In this case, only the positive sequence circuit is considered for the initialization stage of the simulation. With this in mind, the positive sequence based Matpower *power flow* calculation algorithm is directly applied for the initial computations required at the beginning of the simulation process to determine the steady state network operating conditions. The admittance matrix is formulated for the network solution similar to the symmetrical transient stability method. As shown in Equation (2.30), the network solution during normal and fault conditions is based on the current balance form for each of the three sequence circuits. Nodal admittances matrices are constructed for the positive (Y_{bus1}), negative (Y_{bus2}) and zero (Y_{bus0}) sequence circuits from the respective network data. In addition, the generator internal admittance formed from Equation (2.31) and static load admittance from Equation (2.33) are added to the diagonal components of the respective sequence admittance matrices to form the augmented network nodal admittance matrices for each circuit.

Definition of the *event file* component is similar to that used in the symmetrical transient stability method. However, the modified *event file* component in the three-phase transient stability analysis function includes an input for the type of fault to be analyzed and additional parameters for defining the required changes in each of the sequence networks to represent a disturbance in the system. In addition to the balanced events, the fault types defined in the event file include single line-to-ground fault, double line-to-ground fault, line-to-line fault, and three phase-to-ground fault.

The *dynamic component* block is similar to that used in the TS method. Modifications in the mathematical modeling of the dynamic components in the 3Ph-TS method are described in Section 2.3.2. Compared to the TS method, the main differences between the models in the two methods are seen in the algebraic equations of the generator to represent the effects of negative and zero sequence circuit as a result of network imbalances. Moreover, additional parameters are defined in the *dynamic casefile* for the sequence generator reactances used in the formulation of the generator dynamic and algebraic equations and the admittance matrices for the respective sequence circuits.

As a final block is the *DAE solver* component, whose structure is similar to that applied in the TS method, but includes additional functions for the solution of the modified models in the

3Ph-TS method. The numerical integration solvers for the differential equations and the linear algebraic solvers for the network equations are similar to the ones implemented in the basic program of the TS method. However, the solution method is modified in the fault handling process since the 3Ph-TS functions specifically applies the symmetrical components technique for the analysis of symmetrical and asymmetrical transients. For this reason, a new function is included in the solver component for solving the three decoupled sequence circuits during normal operation and for interconnecting the three circuits during the event handling process. The numerical solution process and the event handling function of the modified function are explained in detail in the following subsection.

3.2.3 Solution Methodology

The solution process in the developed three-phase transient stability computation method starts from steady state operation and a network disturbance is applied at a predefined time point during the simulation. For the developed analysis method, the system is also assumed to start from a balanced network operating condition. Thus, only the single phase positive sequence network is used for the initial computations to initialize the network to steady state operations. As described in Section 3.2.2, the Matpower toolbox is applied for the power flow computations. The power flow problem is formulated using an algebraic equation of the form

$$g_z(z) = 0 \tag{3.3}$$

consisting of a set of equations relating the power balance at the network nodes to the bus voltages. Detailed explanation for the formulation of the power flow problem and the different solution algorithms in the Matpower toolbox are given in [33]. The computed variables constituting vector z in Equation (3.3) include the bus voltages, in terms of magnitude and angle, and the active and reactive power generation at the buses. The results of the power flow calculation define the initial value z_0 of vector z in Equation (3.3). These values are used in Equation (2.37) to determine the initial values of the rest of the algebraic variables y_0 and the dynamic state variables by solving

$$f(x_0, y_0, u) = 0. \tag{3.4}$$

Note that the computed initial values in this case only refer to the positive sequence network variables since the initial computations are based on the positive sequence network. The initial variables of the negative and zero sequence networks are set to zero in the balanced system considered at the initialization stage of the simulation. The system variables maintain the same steady values if the network structure remains similar to the structure defined at the initialization stage.

From the initial steady state established as described above, the system of differential and algebraic equations in (2.36) and (2.37) is solved using numerical methods at each time step for the dynamic state variables x and algebraic variables y. The initial values x_0 and y_0 at $t = 0$ are used to solve the system in Equations (2.36) and (2.37) for the solution at the initial simulation time step. An alternating approach (also known as partitioned approach) [129] is applied in the numerical solution applied in the presented method. In the alternating approach,

numerical integration techniques are used to solve the differential equations separately for the dynamic state variables x and the algebraic equations are solved separately for y [129]. During the solution for x from Equation (2.36), estimated values of y from the previous time step are used. The new estimated solution of x is then used in Equation (2.37) to determine the improved solution for y. For the discretization step in the solution process, explicit integration methods are used to numerically integrate the differential equations. The numerical solvers used in this developed computation method are adapted from the symmetrical transient stability analysis function based on MatDyn, with details given in [40, 127].

The solution of the algebraic equation mainly consists of the network equations. As noted in Chapter 2, the generator stator and system loads are included as part of the network. Node admittance matrices are formed for the three sequences using the respective bus and branch information from the network definition. In solving the network equations, the original node admittance matrices are augmented by adding the generator and the equivalent load admittances to the diagonal components. This is carried out at every occurrence of a system transient to rebuild the admittance matrix. The considered simplifications in load modeling and generator stator representation as described in Chapter 2 result in a linear network algebraic equation that can be solved using the sparsity-oriented triangular factorization direct linear solver [128]. The solution of the three network equations is also based on the LU factorization method from the Matlab library.

Event Handling

The network structure and admittance matrix vary depending on the simulation period. Before describing the event handling function, it is important to define the applied network admittance for the different simulation periods. Considering a simulation starting at $t = 0$ and stopping at $t = t_{end}$, with a fault applied at $t = t_{fault}$ and cleared at $t = t_{clear}$, the network connections and admittance matrices used in computing the transient behavior are varied as follows:

- From the start of the simulation to $t = t_{fault}$, (i.e. $0 < t < t_{fault}$), the transient computations are carried out using the pre-fault admittance matrix using the positive sequence network.
- During the fault period $t_{fault} \leq t \leq t_{clear}$, the computation is carried out using the fault-on admittance matrices of the three sequence.
- From $t = t_{clear}$ to the end of the simulation at $t = t_{end}$ (i.e. $t_{clear} < t \leq t_{end}$), the post-fault admittance matrix is applied for the solution and only the positive sequence network is considered.

Event handling starts at the instant when the network event is detected. At this point, the admittance matrices of the three circuits are reformulated to account for the change in network structure. The sequence components are connected using the well-known analysis techniques depending on the fault type to be analyzed. Detailed analysis of the network connections and derivation for the solution of network sequence and phase variables are given in [125]. The network connections are summarized as follows:

- Single line-to-ground fault – the three networks are connected in series at the fault point,
- Double line-to-ground fault – the positive network is connected in series with a parallel combination of the negative and zero networks,
- Line-to-line fault – the positive and negative sequence networks are connected in opposite directions such that the positive sequence fault current is the opposite of the negative sequence current,
- Three-phase short circuit fault – is a balanced fault and therefore only the positive sequence network is considered.

The solution at the instant of fault occurrence proceeds with the computation of the algebraic variables since the dynamic states do not change instantaneously. The algebraic variables y after the event are computed by solving the algebraic equation $0 = g_{new}(x, y, u)$, where g_{new} is the new algebraic function resulting from the change in network configuration. The values of x used during the solution of g_{new} refer to the pre-event state values x_{old}. The algebraic variables y_{new} and the pre-event dynamic states x_{old} are used as the initial conditions for determining the new dynamic states x_{new} during the transient period from $\dot{x}_{new} = f(x_{new}, y_{new})$. The event handling process is repeated at every occurrence of an event.

Algorithm 2 : Workflow of the three-phase transient stability analysis method

Inputs: Network structure data – positive, negative and zero sequence
Load event file
Load dynamic casefile
Compute power flow – positive network
Formulate admittance matrices Y_{bus0}, Y_{bus1}, Y_{bus2}
Initialize system to steady state $0 = g(x_0, y_0)$ and $0 = f(x_0, y_0)$
while $t < simulation\ time$ **do**
    Solve DAE: $\dot{x} = f(x, y)$ and $0 = g(x, y)$
    if *event happened* **then**
        Reformulate Y_{bus0}, Y_{bus1} ,Y_{bus2}
        Interconnect networks
        Solve $0 = g_{new}(x_{old}, y_{new})$
    Update: x and y
    Save x_t and y_t variables at time t
    $t \leftarrow t + dt$
    Repeat: Until set simulation time is reached
Return: System variables x and y

Algorithm 2 summarizes the workflow modification in the developed three-phase transient stability computation method. The system state of stability is analyzed in a similar way as in the symmetrical transient stability analysis function using plots of the system variables. In addition to analysis of the generator rotor speed, angle, and power, the three individual phase

and sequence network quantities are analyzed in the three-phase transient stability analysis function.

3.3 Simulation and Evaluation

The current section presents simulation results using the developed time-domain computation method. The aim of the simulations is to validate the developed component models and evaluate the level of accuracy of the developed computation method in capturing the system transient responses. In order to validate the developed computational method, the simulation results are compared to the results obtained using a commercial grade software package DIgSILENT PowerFactory. Furthermore, the developed method is tested using a large network to validate its reliability in analyzing complex network structures. The computational runtime at the different stages of the simulation is further analyzed as a quantitative measure of the performance of the time-domain simulation method for network structures with different levels of complexity.

Standard test networks, ranging from IEEE test systems to larger networks representing the size and complexity of the European power grid [130, 131], are used for presented simulation results. The input files of the network structures are obtained from Matpower casefiles. Table 3.1 gives a summarized description of the networks in terms of the number of generator units (Gens), network buses (nodes), interconnected transmission lines including transformers (branches) and load data in terms of active power (P) and reactive power (Q) consumption. The tested events in the current section are limited to a single line-to-ground fault and a three-phase short circuit fault.

Table 3.1: Summary of standard test network structures

Network case	Number of components			Load data		
	Gens (m)	Nodes (n)	Branches (b)	Loads	P (MW)	Q (MVar)
Case9	3	9	9	3	315.0	115.0
Case30	6	30	41	20	189.2	107.2
Case118	54	118	186	99	4242.0	1438.0
Case300	69	300	411	201	23525.8	7788.0
Case1354pegase	260	1354	1991	673	73059.7	13401.4
Case9241pegase	1445	9241	16049	4895	312354.1	73581.6
Case13659pegase	4092	13659	20467	5544	381431.9	98523.4

3.3.1 Accuracy Validation

The test and benchmarking scenario used to analyze the accuracy of the developed method is based on the standard IEEE nine-bus test feeder (Case9 in Table 3.1). A small network is selected for purposes of illustration. The structural representation of the IEEE 9-bus network is given in Appendix A.1. The synchronous machines at buses 1, 2 and 3 are connected with excitation and turbine-governor controller systems. The general parameters of the test network are

derived from [18]. For the test network in DIgSILENT PowerFactory, the controllers connected to generation units are the excitation system represented by the IEEE type 1 model (IEEE T1) as described in [120], and turbine-governor system represented by the simplified model (TGOV1) as described in [121].

A single line-to-ground fault on phase-A – representing an asymmetrical transient – with zero fault impedance is considered for the benchmarking scenario to validate the accuracy of the simulation method. The fault is applied on bus 6 at time $t = 5\,\mathrm{s}$ and cleared at $t = 15\,\mathrm{s}$. For illustration purposes, a long fault duration of $10\,\mathrm{s}$ is set in the current test case in order to qualitatively analyze the simulation accuracy in terms of the generated profiles of the system variables during the fault and post-fault period. The simulation is run at a fixed step size of $10\,\mathrm{ms}$ for a duration of $30\,\mathrm{s}$.

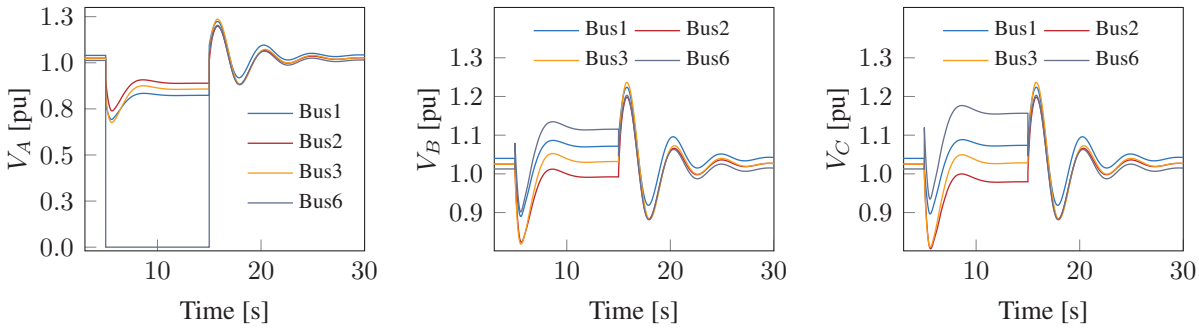
Evaluation of Node Voltage Response

Figures 3.2 and 3.3 show the comparison of the phase and sequence bus voltage responses for the simulated fault scenario using the developed 3ph-TS method and DIgSILENT PowerFactory. Figure 3.2a shows the bus phase voltage response captured by the 3ph-TS method, illustrating the transient behavior for the fault duration and during the post-fault period. An important characteristic of the 3ph-TS method is that all the information of the individual phases (A, B, and C) is available during analysis as observed in Figure 3.2a. The voltages in the three phases are initially in steady state and equal in magnitude. Since the simulated fault is asymmetrical, the responses of the three phase voltages are observed to be differ. At the fault bus, the voltage on phase-A drops, whereas the phase-B and phase-C voltages increase as expected. The voltage drop at fault bus 6 is up to zero, reflecting the zero fault impedance applied in the tested fault scenario. A similar response in the bus phase voltages is observed in DIgSILENT PowerFactory as shown in Figure 3.2b. When the fault is cleared, the initial network structure is reestablished and the action of the connected voltage regulators (excitation systems) results in the voltages returning to the initial operating levels as observed in the post-fault period of the captured responses in both simulation tools.

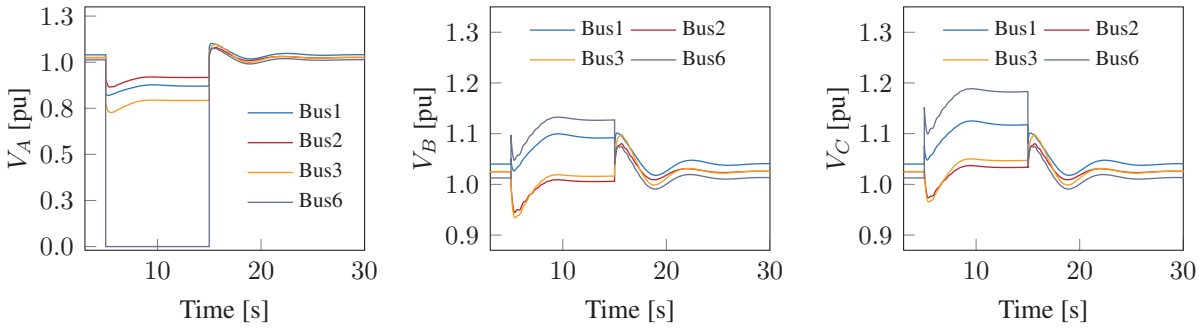
Additionally, the three-phase transient stability analysis method derives the sequence quantities of the system variables. Figure 3.3 shows the responses of the sequence voltages obtained using the 3ph-TS method and in DIgSILENT PowerFactory. It is observed that only the positive sequence voltage exists during normal operation under balanced condition. The negative and zero sequence voltages result from the network imbalance due to the applied asymmetrical fault. The 3ph-TS method has demonstrated the ability to derive the individual phase and sequence information. This kind of information cannot be obtained in the traditional transient stability analysis tools in which the representation is based on the positive sequence network .

Evaluation of Machine Speed and Power Response

The comparison of the machine rotational speed responses is shown in Figure 3.4. Initially, the machines are in synchronism with a speed of 1.0 pu. Following the fault at $5\,\mathrm{s}$, the change in network topology results in acceleration of the machine rotors as seen in Figure 3.4 during the fault period. The change in rotor speed initiates the action of the turbine-governor systems,



(a) Phase-voltages in the proposed Matlab-based method



(b) Phase-voltages in DIgSILENT PowerFactory

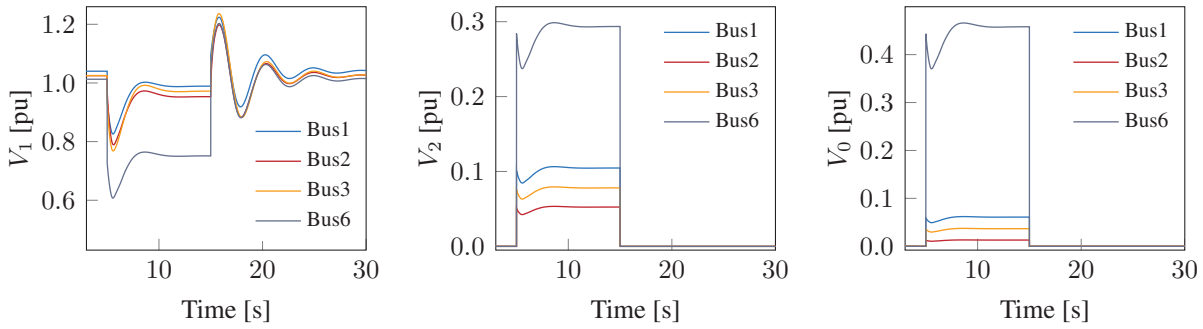
Figure 3.2: Comparison of Phase-A,-B, and -C voltage responses for a single-phase (Phase-A) short circuit fault in the proposed Matlab-based method and DIgSILENT PowerFactory

which adjust the mechanical power contribution of each machine as shown in Figure 3.5; thus reducing the speed deviation until synchronous operation of the interconnected machines is achieved. The responses of the generator rotational speed and turbine-power in the 3ph-TS function are observed to closely match the responses in DIgSILENT PowerFactory.

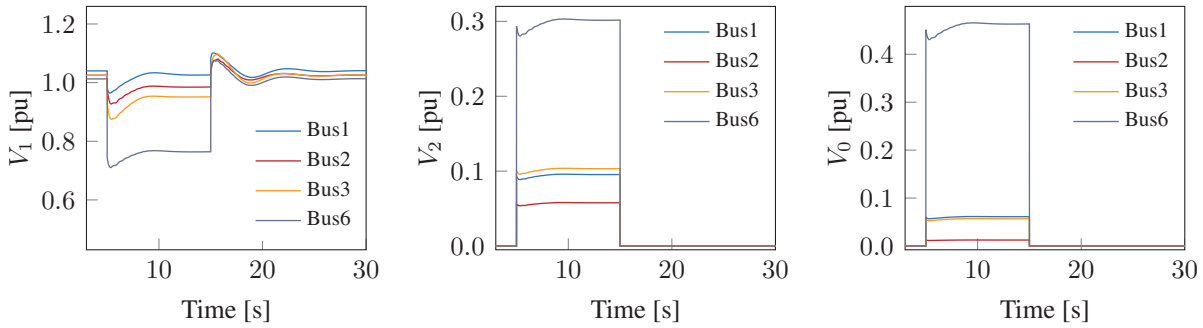
Figures 3.2 to 3.5 show the response of the bus voltages, generator rotational speed and power obtained using the 3ph-TS method in comparison to DIgSILENT PowerFactory. The noticeable difference in the response following the change in the network topology can be attributed to the inherent differences in the model details and implementation of the numerical solvers in the software packages. Nevertheless, qualitative analysis of the results shows a close match in the responses in the two software packages. Therefore, the proposed simulation method described in the current chapter is able to analyze system transients with an accuracy level comparable to a validated commercial software package.

Validation in 9241-Bus Network

The aim of the simulations in the current section is to analyze the ability of the developed computational method to handle large and complex networks. For this purpose, the simulation method is tested using a 9241-bus network (Case9241pegase in Table 3.1) representing the size and complexity of the European high voltage network [131]. In this test case, three-phase faults

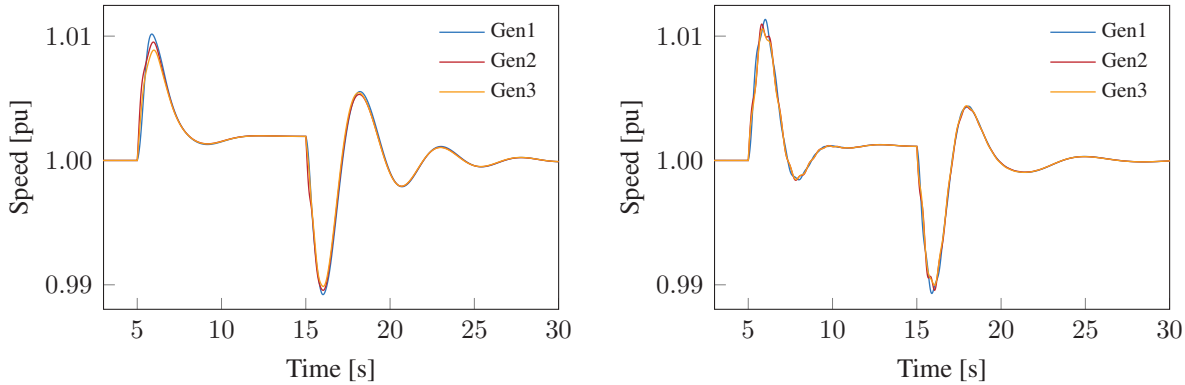


(a) Sequence-voltages in the proposed Matlab-based method



(b) Sequence-voltages in DIgSILENT PowerFactory

Figure 3.3: Comparison of positive-, negative- and zero- sequence voltages for a single-phase (Phase-A) short circuit fault in the proposed Matlab-based method and DIgSILENT PowerFactory

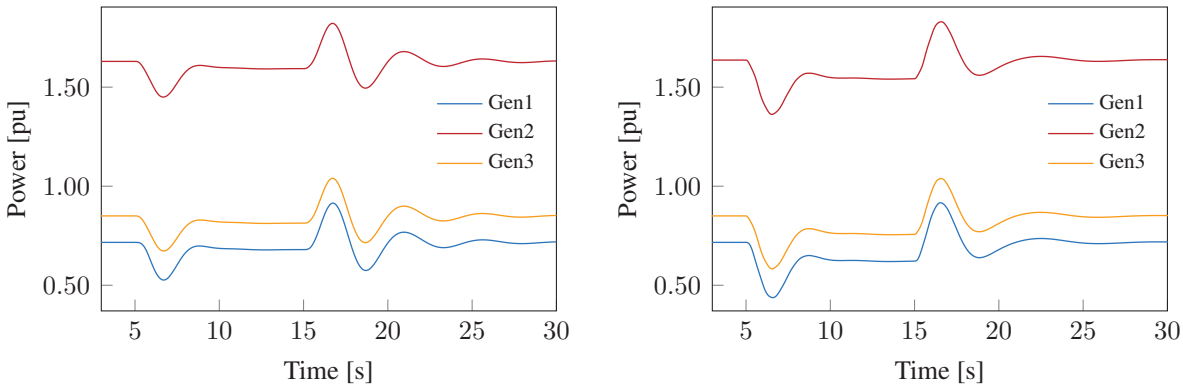


(a) Speed in proposed Matlab-based method

(b) Speed in DIgSILENT PowerFactory

Figure 3.4: Comparison of rotational speed response in the proposed Matlab-based method and DIgSILENT PowerFactory

are applied at two separate network locations at different times during the simulation. The first fault is applied on bus 28 at 3 s and cleared after 100 ms. The second fault is applied on bus 143 at 5 s and cleared after 100 ms. The results of the generator rotational speed and bus voltage response obtained using the 3ph-TS method are shown in Figure 3.6a and Figure 3.6b,

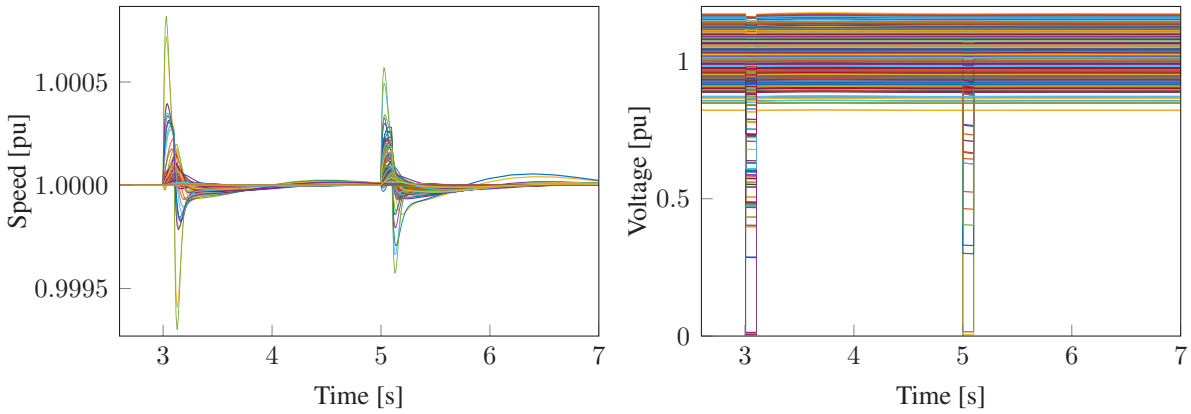


(a) Power in proposed Matlab-based method

(b) Power in DIgSILENT PowerFactory

Figure 3.5: Comparison of mechanical power response in the proposed Matlab-based method and DIgSILENT PowerFactory

respectively. Since the applied disturbance is a balanced fault, the voltage responses in the three phases are similar. Thereby the bus voltage response can be presented on a single phase basis as shown in Figure3.6b.



(a) Generator rotational speed response

(b) Bus voltage response

Figure 3.6: Generator rotational speed and bus voltage in response to two faults on a 9241-bus network obtained using the proposed Matlab-based method

The results obtained from the simulation of the 9241-bus network accurately represent the expected behavior of the bus voltage and generator rotational speed in response to a balanced network fault. Therefore, the developed method is capable of handling the complexity involved in the analysis of network transients in large networks.

3.3.2 Assessment of Computational Complexity

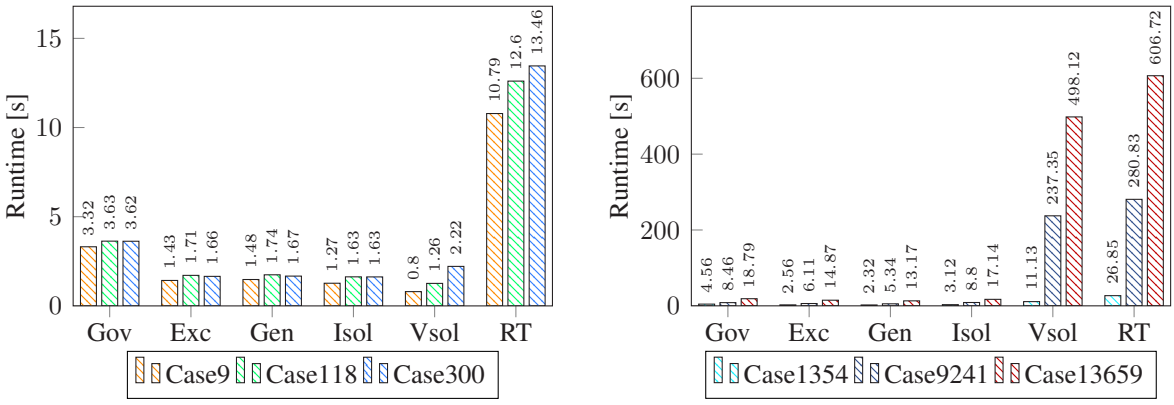
In the current section, the computational complexity of the simulation algorithm is analyzed for varying network sizes. The main objective for the tests is to identify the most expensive tasks of

the computational method in relation to the complexity of the network structure. The computer platform used for the simulation tests is Intel(R) Core(TM) i7-8700 CPU @ 3.20 GHz, 32 GB memory, Microsoft Windows 10.

For each network as described in Table 3.1, the computation runtime is analyzed at the following dominant stages of the simulation algorithm: Solution of the differential and algebraic equations of the generator subsystem for the governor system – Gov, exciter system – Exc, and generator system – Gen; Calculation of the node current injection – $Isol$; Solution of the network equation for the node voltages – $Vsol$. The total computation time – RT is the time taken for the execution of the whole simulation.

Figures 3.7a and 3.7b show the summary of the computation runtime for the dominant stages of the algorithm for the different network structures. For illustration purposes, networks with the number of nodes n less than 1000 nodes ($n < 1000$; Case9, Case118 and Case300) are shown in Figure 3.7a and the networks with n greater than 1000 ($n > 1000$; Case1354, Case9241 and Case13659) are shown in Figure 3.7b. Figure 3.7a shows that the most expensive task for small networks is the solution of the differential equations and associated algebraic equations in the generator subsystem, dominated by the solution of the governor equations. Combined together, the solution of the generator subsystem equations contributes to about 60% of the simulation runtime for the small networks. The solution of these equations mainly involves vector operations and therefore have an estimated linear computational complexity, which depends on the number of generators. The computational complexity expressed in the $O-$notation is $O(m)$ for m generators in the system.

On the other hand, as the number of network nodes n increases, the most expensive task becomes the solution of the network equations as shown is Figure 3.7b. Solving the linear system of equations by the forward and backward substitution has a computational complexity of $O(n^2)$ and therefore grows faster than the vector operations involved in the solution of the generator subsystem equations. As observed from Figure 3.7b, the network solution takes up about 80% of the simulation runtime for the large networks.



(a) Networks with nodes n less than 1000

(b) Networks with nodes n greater than 1000

Figure 3.7: Computational runtime of the main stages of the time-domain simulation algorithm

3.4 Summary

The main contribution of the current chapter is a computational method for analyzing network transients considering symmetrical and asymmetrical power system operating conditions. The method proposes a considerable improvement in power system analysis, especially in the current operating environment where unbalanced network operations can no longer be overlooked during system analysis. Test simulations have shown the accuracy the method to be comparable to that of a commercial grade simulation package DIgSILENT PowerFactory. With the presented results, the current chapter addresses the open questions regarding the need for flexibility and functionality extension in power system analysis methods. Additional tests using large networks have shown the ability of the method to handle analysis of transients in large networks. The main challenge when dealing with large networks is the large computational time required for the simulations. The runtime of the algorithm is observed to increase with a polynomial complexity as the size of the network increases. The qualitative profiling of the different stage in the simulation algorithm provides insight into which stages should be considered to address the computational complexity in time-domain simulations.

However, the network models in the current chapter have considered only conventional generation sources. The objective of including models of renewable energy sources in system analysis is addressed in Chapter 4 of the present thesis. Furthermore, the objective of addressing the computational complexity identified with the time-domain simulation method is covered in Chapter 5 using grid partitioning and parallel computing techniques.

4 Time-Domain Simulation of Power System with Renewable Energy Sources

The components considered during modeling and analysis in the present thesis until now are the standard components of the conventional power system. However, the energy transition towards carbon-free energy generation has led to integration of more renewable energy sources that introduce different dynamic characteristics to the power system operation. In the present chapter, models of solar photovoltaic and wind power generators are developed to include renewable energy sources into the overall system analysis. Specifically, the models are developed in order to analyze the impact of large-scale integration of renewable energy sources. Among the important factors to assess is the impact on system stability if conventional generators – based on synchronous machines with high inertia – are replaced with variable renewable generators. Additionally, it is necessary to evaluate the required measures to maintain the network frequency within normal operational limits in the presence of continuous fluctuations in power generation.

Given the inherent connection of photovoltaic and wind power generators to the grid through power electronic converters, new high-level converter functions are developed in the current chapter for interfacing the renewable energy generators to the electrical network in reference to standard grid operation codes. The models and corresponding functions are integrated into the time-domain computational framework and tested in order to analyze the different control strategies necessary for system stability support.

4.1 Dynamic Modeling of Photovoltaic Generation System

A grid connected PV generation system is of interest in the present thesis, where the generated power is injected into the grid. Figure 4.1 shows the main components of the PV generator system for a single stage conversion system. In this system, the power converter interface constitutes the conversion stage and is used as an interconnection between the PV arrays and the power grid. The most important component of the power converter interface is the inverter,

---

The content of this chapter is an extension of the work presented in the publications [132, 133].

M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Dynamic Modeling of Wind and Solar Power Generation with Grid Support for Large-Scale Integration in Power Systems," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, The Hague, NL, pp. 569–573, October 2020.

M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Dynamic Modeling and Control for Assessment of Large-Scale Wind and Solar Integration in Power Systems," *IET Renewable Power Generation (In print)*, 2020.

which converts the PV array DC output voltage into AC voltage and performs the necessary grid interconnecting functions [134]. The inverter settings are defined according to specified requirements as set by network regulations for power plants [135]. These network regulations are the basis of the inverter model developed in the present thesis.

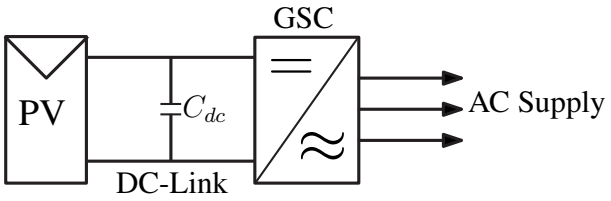


Figure 4.1: PV interface topology using single-stage conversion with Grid-Side Converter (GSC) [134]

4.1.1 Simplifying Assumptions

A complete PV system model includes components for computation of the solar radiation, PV array power production, maximum power point tracking (MPPT), power converter, and converter controllers. In view of the wide range of response times for the different PV system components, the component models are conveniently divided into steady state and dynamic models. Steady state models account for long-term effects and consider variables that are affected by variations in solar radiation and cell temperature. Since the main focus of the current thesis is on the response of the power system to short-term transients, the functions modeling the behavior of long-term steady state PV components are not considered in the presented models. A description of the PV system steady state analysis functions is given in [136].

Dynamic components of the model are derived to study short-term effects on a time scale of a few seconds. The main assumption considered in deriving the dynamic models in the present thesis is that the long-term variables, such as solar radiation and the maximum power point, are constant for the study period of the system dynamics according to the time-scale separation based on the singular perturbation theory [137]. This simplifies the model of the PV array for the analysis of dynamics to a constant power source. The constant value of the PV power output is determined directly from steady state calculations at a specific time step.

4.1.2 DC-Link Model

The current section describes the model of the DC side of the inverter, i.e. the DC-link. As shown in Figure 4.1, the DC-Link is the interface between the PV-array and the grid-side converter (inverter). The DC-Link is modeled under the assumption that the stationary maximum power point does not change for the short time frame considered during the analysis of system dynamics. This implies that the DC power input P_{dc} to the inverter at the start of the simulation is equal to the PV array power output P_{pv} and the maximum power point P_{mpp}. Equation (4.1) shows the general condition for the steady state initialization of the PV model.

$$P_{ac} = P_{inv} = P_{dc} = P_{mpp} = P_{pv} \tag{4.1}$$

In Equation (4.1), P_{ac} is the inverter active output power and P_{inv} is the inverter input power. The derived model is based on the single stage conversion system as shown in Figure 4.1. The voltage at the inverter input is determined predominately by the capacitor connected in the DC-link. The dynamics of the DC-link voltage are described from the relationship giving the interaction between the input power P_{dc} and the output power P_{inv} of the DC-link given by [134]

$$\frac{dV_{dc}}{dt} = \frac{P_{dc} - P_{inv}}{C_{dc}V_{dc}} \tag{4.2}$$

where V_{dc} is the DC-link voltage and C_{dc} is DC-link capacitance. The PV array power P_{pv} is directly dependent on the array voltage. In the single-stage conversion system considered here, the array voltage is directly connected to the inverter. During short-term variations, the dynamic behavior of the inverter DC side depends on the control of the inverter active current.

4.1.3 Grid-Side Converter

The grid-side converter is an active inverter used to convert the DC-link voltage to AC-grid voltage at a fixed frequency of the power grid. Grid operation codes require inverter-connected generation sources to actively contribute to grid stability. Unlike the conventional synchronous generators described in Chapter 2, whose dynamic behavior is due to their physical construction, the dynamic response of inverter-connected generators is defined by control functions of the inverter. The new functions developed for the grid-side converter model according to grid operation standards are described in Section 4.3.

4.2 Dynamic Modeling of Wind Power Generation System

A variable speed wind turbine with a direct-drive synchronous machine system is considered in the wind power generation system described in the present thesis. Further details of the components and operation of the commonly used configurations in wind power systems – which include fixed speed wind turbine with a squirrel cage induction generator, variable speed wind turbine with a doubly-fed induction generator, and variable speed wind turbine with a direct-drive synchronous generator – are given in [23]. Figure 4.2 shows the wind generator system configuration considered in the derived model consisting of a wind turbine, synchronous machine (SG), and power converter (AC/DC – rectifier, and DC/AC – inverter converters).

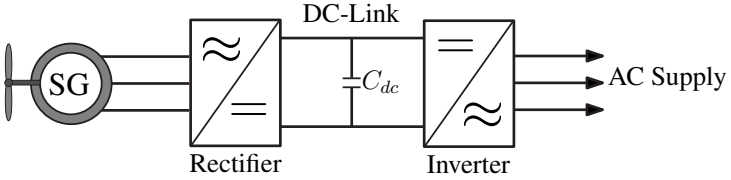


Figure 4.2: Schematic representation of wind power generator with a fully-rated converter system

4.2.1 Simplifying Assumptions

In the current thesis, modeling of the wind turbine is simplified since only the electrical behavior of the system is of interest. The mechanical power P_w extracted from the wind is a function of the wind speed v_w, the rotor rotational speed ω_t, the pitch angle θ_p, and the turbine power coefficient characteristic c_p [138]. From the steady state calculations as described in [139], the variation of mechanical power output with turbine rotational speed can be computed using a defined c_p characteristic. The values of the mechanical power and the rotational speed from the turbine stage are the inputs to the generator stage. Since the wind speed does not change drastically in a few seconds, the mechanical power is assumed to remain constant for the duration of the dynamics simulation.

The stator windings of the synchronous generator are connected to the power grid through a full-scale frequency converter. The full-scale frequency converter consists of a generator-side converter, a DC-link, a grid-side converter, and the corresponding controllers. The converter used in the model described in the current section is a self-commutated current source converter (CSC) based on pulse width modulated (PWM) Insulated-Gate Bipolar Transistor (IGBT) technology. The advantage of using a full-scale converter is that the generator is completely decoupled from the grid. This simplifies the converter functions and particularly offers better regulation of reactive power, especially during grid faults, which is important for grid stability [95]. The models of the individual short-term dynamic components of the wind power generation system are described in the following subsections.

4.2.2 Synchronous Generator

The synchronous generator used in direct-drive systems can be a wound-field or a permanent magnet machine. In the present thesis, a permanent magnet synchronous generator (PMSG) is selected due to its characteristic advantage of eliminating the need for a DC excitation system and slip rings, resulting in reduced losses and less maintenance requirements [140]. The PMSG therefore attains a higher efficiency compared to the wound-field machines. The dynamic model of the PMSG as applied in current thesis is described as follows: The stator voltage equations in the rotor flux reference frame are given by

$$\begin{aligned} v_{ds} &= -R_s i_{ds} - \omega_s \psi_{qs} + \frac{d\psi_{ds}}{dt} \\ v_{qs} &= -R_s i_{qs} + \omega_s \psi_{ds} + \frac{d\psi_{qs}}{dt} \end{aligned} \tag{4.3}$$

where v_{ds} and v_{qs} are the d- and q-axis components of the stator terminal voltage; i_{ds} and i_{qs} are the d- and q-axis components of the stator current; R_s is the stator resistance; ω_s is the generator electrical angular speed given by $\omega_s = p\omega_m$ as a function of the mechanical angular speed ω_m and number of generator pole pairs p. The d- and q-axis stator flux linkages ψ_{ds} and ψ_{qs} are calculated from [95]

$$\begin{aligned} \psi_{ds} &= -L_{ds} i_{ds} + \psi_{pm} \\ \psi_{qs} &= -L_{qs} i_{qs} \end{aligned} \tag{4.4}$$

whereby L_{ds} and L_{qs} are the respective d- and q-axis stator leakage inductances, and ψ_{pm} is the permanent magnet flux leakage. The generator active power P_g and reactive power Q_g are obtained according to Equation (4.5).

$$\begin{aligned} P_g &= \frac{3}{2}(v_{ds}i_{ds} + v_{qs}i_{qs}) \\ Q_g &= \frac{3}{2}(v_{qs}i_{ds} - v_{ds}i_{qs}) \end{aligned} \tag{4.5}$$

The mechanical shaft of the generator is modeled using the equation of motion. In the direct-drive system, the full-scale frequency converter is assumed to filter out the shaft dynamics. Thereby, all rotating masses are represented by an equivalent single shaft in the model. The resulting generator mechanical equation is

$$\frac{d\omega_m}{dt} = \frac{1}{2H_m}(T_m - T_e) \tag{4.6}$$

where H_m is the generator rotor inertia and T_m is the generator mechanical torque. The electrical torque T_e in Equation (4.6) is given by

$$T_e = \frac{3}{2}p\Big[\psi_{pm} + (L_q - L_d)i_{ds}\Big]i_{qs} \tag{4.7}$$

The voltage at the shunt capacitor C_r connected across the generator terminals is given by Equation (4.8) in the dq reference frame.

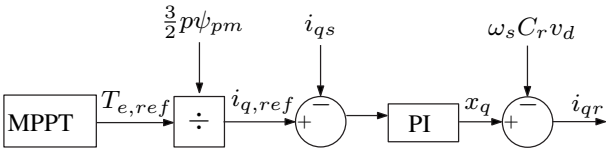
$$\begin{aligned} \frac{dv_{ds}}{dt} &= \omega_s v_{qs} + \frac{i_{ds}}{C_r} - \frac{i_{qr}}{C_r} \\ \frac{dv_{qs}}{dt} &= \omega_s v_{ds} + \frac{i_{qs}}{C_r} - \frac{i_{dr}}{C_r} \end{aligned} \tag{4.8}$$

4.2.3 Generator-Side Converter Control

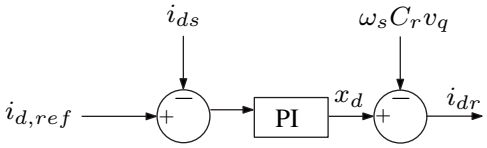
As described in the previous section, the generator model is based on a permanent magnet synchronous generator. Therefore, the generator has a fixed excitation. In order to model the generation system with a degree of controllability to achieve optimal performance, the PMSG system is modeled with controllable valves. In this case, the generator-side converter is represented by a fully controllable active PWM controlled IGBT converter. However, since the analysis in the present thesis in based on the fundamental frequency assumptions, the high switching frequency of the IGBT power electronics are not considered following the time-scale separation [137]. The converter is therefore modeled using an average model, which models the dynamics of the converter control systems but neglects the high frequency switching dynamics of the IGBTs.

The derived converter controller model is based on the full torque control strategy [140, 141]. In this strategy, the total stator current is induced in the q-axis of the stator and the d-axis current is set to zero. From Equation (4.7), this results in maximum torque generation from

the generator. The converter control is split into the d- and q-axis current control as illustrated in Figure 4.3. The reference active power is determined from maximum power point tracking (MPPT) as a steady state control function. The controlled quantities by the generator-side converter are the generator active power and terminal voltage using the converter currents.



(a) q-axis current control



(b) d-axis current control

Figure 4.3: Generator-side converter control scheme; (a) q-axis current control with maximum power point tracking (MPPT); (b) d-axis current control with zero direct-axis current control ($I_{d,ref}$)

The control of the q-axis loop is given by

$$\begin{aligned} i_{q,ref} &= \frac{2T_{e,ref}}{3p\psi_{pm}} \\ \frac{dx_q}{dt} &= K_{i,q}(i_{q,ref} - i_{qs}) \\ i_{qr} &= x_q + K_{p,q}(i_{q,ref} - i_{qs}) - \omega_s C_r v_{ds} \end{aligned} \tag{4.9}$$

The d-axis control loop is modeled in such a way that the value of the d-axis current is regulated to a reference current $i_{d,ref} = 0$. The resulting controller equation is

$$\begin{aligned} \frac{dx_d}{dt} &= K_{i,d}(i_{d,ref} - i_{ds}) \\ i_{dr} &= x_d + K_{p,d}(i_{d,ref} - i_{ds}) + \omega_s C_r v_{qs} \end{aligned} \tag{4.10}$$

In Equations (4.9) and (4.10), K_i and K_p are the PI-controller's integral and proportional gains of the corresponding control loops; x_d and x_q are intermediate state variables within the control loops; the output signals i_{qr} and i_{dr} of the q- and d-axis control loops are used to control the pulse width modulated IGBTs of the rectifier. The feed forward terms $\omega_s C_r v_{ds}$ and $\omega_s C_r v_{qs}$ in Figure 4.3 are added to compensate for the cross coupling between the d- and q-control loops, which results from the transformation of the currents from the network reference frame to the rotor rotating reference frame.

Controller Tuning

An additional requirement for the model of the generator-side converter is the right tuning of the PI-controller parameters. In the derived system model, the proportional gain and the integration time constant of the controller are estimated based on the optimum value estimation strategy described in [142] as follows: The general transfer function of a PI-controller is expressed as

$$G_{PI}(s) = K_p + \frac{K_i}{s} = K_p\left(1 + \frac{1}{sT_i}\right) \tag{4.11}$$

If the dynamics of the controlled generator system in Equation (4.3) are expressed as a first order transfer function of the form $G_1(s) = (\alpha)/(1 + s\beta)$ and the generator-side converter (rectifier) is represented by a simplified first order transfer function with a time constant γ as $G_2(s) = 1/(1 + s\gamma)$, the resulting controlled system is then given by

$$G(s) = \frac{\alpha}{(1 + s\beta)(1 + s\gamma)} \tag{4.12}$$

The controlled system is therefore represented by a second order transfer function as shown in Equation (4.12). From the optimization table giving the relation between the controlled system and controller settings defined in [142], the values of K_p and T_i in Equation (4.11) are defined from Equation (4.12) as

$$\begin{aligned} T_i &= \beta \\ K_p &= \frac{\beta}{2\alpha\gamma} \end{aligned} \tag{4.13}$$

whereby the variables α, β are known from the modeled controlled system in Equation (4.3), and γ is a known time constant for the first order transfer function of the rectifier.

4.2.4 DC-Link

The DC-link decouples the generator-side converter from the grid-side converter. As shown in Figure 4.2, the DC-link is composed of a capacitor C_{dc}, which provides the intermediate energy storage and regulates the link voltage. Assuming a lossless converter system, the dynamics of the DC-link are described by

$$\frac{dV_{dc}}{dt} = \frac{P_g - P_{inv}}{C_{dc}V_{dc}} \tag{4.14}$$

where V_{dc} is the DC-link voltage, P_g is the generator-side active power and P_{inv} is the inverter-side output power.

In addition to the energy storage element, the DC-link contains a braking chopper circuit which serves as a protection component for the DC-link by dissipating excess power through a braking resistor. The chopper is triggered during conditions when the wind energy system cannot inject all the active power into the grid, for example during the fault ride-through operation to support the grid. The DC-link discharges through the chopper circuit to keep the link

voltage below a critical value and enable fault ride-through capability. In this way, the generator is operated at the optimal point during fault, and can continue to provide the necessary grid support functions without overloading the DC-link element. The developed model for the chopper circuit is given by a first order transfer function approximation as

$$\Delta P_{chp} = (P_g - P_{inv})\left(1 - e^{-\frac{\Delta t}{\tau}}\right) \tag{4.15}$$

where ΔP_{chp} is the change in power of the chopper, Δt is the time step, and τ is the chopper regulation time constant.

4.2.5 Grid-Side Converter Control

Considering the electrical response and neglecting the effects of the energy source on the grid, the characteristics of the wind generator interface are similar to those of a PV generator interface. Therefore, the new grid-side converter model described in Section 4.3 is applied for both the PV generation and wind power generation systems. Similar inverter functions are applied for control of the power injected into the grid and the grid-side voltage.

4.3 New Inverter Control Model

The inverter model described in the present section is a current source inverter. Modeling of the inverter control is based on the principle that inverter regulation depends on its actual functionality. The variables used for the regulation are the current and voltage at the output terminals of the inverter. The dynamic response of the inverter is defined by the control functions, which are divided into Low-level and high-level functions. Low-level control regulates the current output by controlling the power electronic switching devices. Considering the time-frame of the dynamics simulation, low-level functions for controlling the fast switching action of power electronic devices are not considered in the present thesis. On the other hand, high-level control functions regulate the power into the grid depending on the grid regulations and specifications. In the current thesis, the following inverter control functions are developed for high-level control: Active power reduction; Reactive power support; and Fault ride-through (FRT).

4.3.1 Active Power Reduction Control Function

The frequency-dependent active power reduction control function of the inverter model is developed to represent the short-term frequency response of an inverter-connected system in reference to defined grid interconnection standards [143]. This function requires the inverter-connected generation system to reduce the active power injection if the frequency deviation is greater than a specified threshold value. During this period, the system operates below the maximum power point. Figure 4.4 illustrates a sample profile followed by a generation system during active power reduction for a 40% power reduction per frequency deviation greater than 0.2 Hz. In case of a change in frequency, either above 51.5 Hz or below 47.5 Hz, the set standards require the PV system to be disconnected from the grid [143].

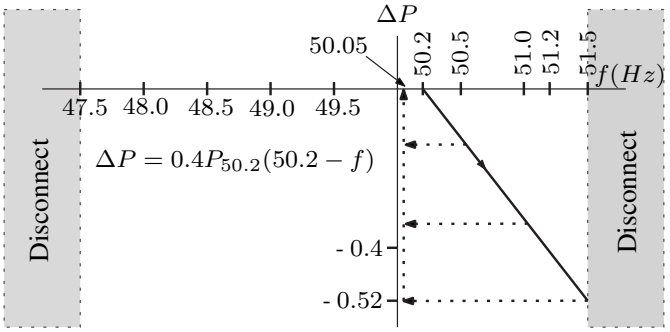


Figure 4.4: Frequency–dependent active power reduction profile [143]

The developed active power reduction control function is based on a first order transfer function simplification. The dynamic frequency f_{bus} at each bus is approximated by

$$f_{bus} = f_0 + \frac{d\theta_{bus}}{dt} \tag{4.16}$$

where f_0 is the synchronous system frequency, and $\frac{d\theta_{bus}}{dt}$ is the derivative of the bus voltage angle. The measured frequency at the inverter terminal is computed from

$$f_{inv,t} = f_{inv,t-\Delta t} + (f_{bus} - f_{inv,t-\Delta t})\left(1 - e^{-\frac{\Delta t}{\tau}}\right) \tag{4.17}$$

In Equation (4.17), $f_{inv,t}$ is the measured frequency at time t, $f_{inv,t-\Delta t}$ is the frequency at the previous time step, Δt is the time step, and τ is the time constant for frequency measurement.

The variation of the inverter current $\Delta I_{d,q}$ in the d- and q-axis is calculated assuming a first order transfer function approximation by

$$\Delta I_{d,q} = (I_{dq,ref} - I_{dq,t-\Delta t})\left(1 - e^{-\frac{\Delta t}{\tau}}\right) \tag{4.18}$$

where $I_{dq,ref}$ is the nominal current of the respective controller, $I_{dq,t-\Delta t}$ is the current at a previous time step, and τ is the controller time constant. The nominal active current value is estimated depending on the frequency range of operation. In the developed inverter functions, the frequency ranges are defined according to the frequency deviation as follows: 1) frequency deviation below a specified threshold; 2) frequency deviation outside the set threshold. In the first case, the active current $I_{d,ref}$ is calculated from the specified power P_{dc} and the inverter bus voltage V_{inv} according to Equation (4.19). In the second case, $I_{d,ref}$ is computed from the actual power P_{act} at the frequency threshold, power deviation ΔP, and voltage magnitude as shown in Equation (4.20). The power deviation in Equation (4.20) is calculated from a defined active power reduction profile according to standards set by network operators [144], as illustrated in Figure 4.4.

$$I_{d,ref} = \frac{P_{dc}}{|V_{inv}|} \tag{4.19}$$

$$I_{d,ref} = \frac{P_{act} - \Delta P}{|V_{inv}|} \tag{4.20}$$

4.3.2 Reactive Power Support Control Function

The reactive power support control function is developed as a static grid support measure for voltage regulation as required by network standards [143]. In practical applications, the required reactive power from an inverter-connected generation system is specified by the grid operator according to either a set value or a characteristic curve depending on the plant operating point [145]. The set value is defined by either a constant reactive power in Mvar or a constant power factor. On the other hand, the characteristic curves for reactive power capability are defined by reactive power – voltage ($Q - V$), reactive power – active power ($Q - P$), and power factor – active power ($\cos\theta - P$) characteristics [143, 146].

In the reactive power support control function, it is assumed that the reactive power estimated from the initial steady state computations can be directly applied for short-term dynamic simulations. To include dynamic grid support measures, the reactive power value can be varied between constant current, given by the initial reactive current I_{q0}, and constant power depending on the initial reactive power Q_0. The corresponding value of the nominal reactive current $I_{q,ref}$ is calculated according to Equation (4.21) for constant current type, and Equation (4.22) for constant power type.

$$I_{q,ref} = I_{q0} \tag{4.21}$$

$$I_{q,ref} = -\frac{Q_0}{|V_{inv}|} \tag{4.22}$$

4.3.3 Fault Ride-Through Control Function

The fault ride-through (FRT) control function describes the ability of the generating system to remain connected to the network and support the grid during voltage sags caused by a fault condition [147, 148]. Grid standards define limit profiles that a generating unit must comply with in supporting the network. Details of a typical fault ride through characteristic with sample lower limits of voltage levels before, during, and after a fault are given in [144].

In the developed FRT function, the regulation of injected current into the grid during fault is divided into two categories: FRT with voltage support, and FRT without consideration of voltage support. For FRT without voltage support, the total injected current is set to zero. On the other hand, additional reactive current is injected into the grid for FRT with voltage support. Figure 4.5 depicts a voltage support activation profile defined by the grid code for inverter-interfaced renewable energy generators during fault ride-through [145].

The injected reactive current is calculated according to Equation (4.23)

$$\Delta i_q = k\Delta v \tag{4.23}$$

$$\Delta i_q = \frac{\Delta I_q}{|I_n|}; \Delta I_q = I_q - I_0 \tag{4.24}$$

$$\Delta v = \frac{\Delta V}{|V_n|}; \Delta V = V - V_0 \tag{4.25}$$

whereby Δi_q is the normalized additional reactive current defined in Equation (4.24), ΔI_q is the

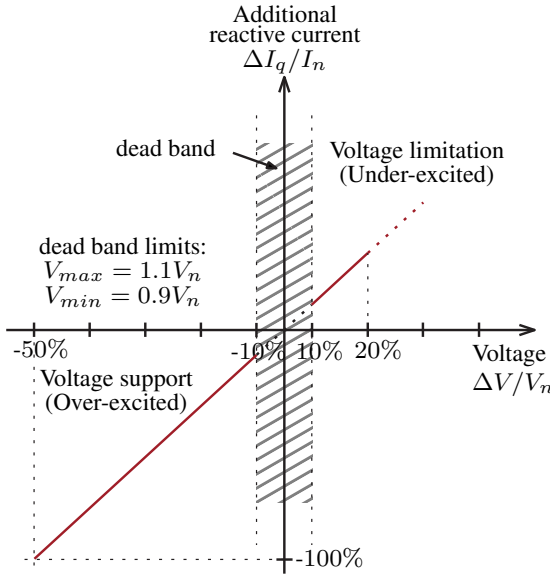


Figure 4.5: Voltage support characteristic during fault ride-through operation for inverter-interfaced renewable energy generators [145]

required additional reactive current, I_q is the actual reactive current, I_0 is the reactive current prior to the fault and I_n is the nominal reactive current. Δv is the normalized voltage deviation given by Equation (4.25), where ΔV is the absolute voltage deviation, V is the actual voltage value during the fault, V_0 is the voltage before the fault, and V_n is the nominal voltage. Parameter k is the reactive current-voltage gain, which represents the percentage increase of reactive current per percentage of the voltage drop [145]. The fault is considered to be present on the network until the voltage settles back to acceptable operational limits within the dead band around the reference voltage.

4.3.4 Inverter Current Limits

Considering the functions derived above, the active and reactive current outputs of the inverter are calculated from the changes due to active power reduction $\Delta I_{d,P}$, fault ride-through $\Delta I_{dq,FRT}$, and static reactive power control $\Delta I_{q,stat}$ as given in Equations (4.26) and (4.27). The total inverter current is calculated from the active and reactive current components according to Equation (4.28).

$$I_{d,new} = I_{d,old} + \Delta I_{d,P} + \Delta I_{d,FRT} \tag{4.26}$$

$$I_{q,new} = I_{q,old} + \Delta I_{q,stat} + \Delta I_{q,FRT} \tag{4.27}$$

$$I_{inv} = \sqrt{(I_{d,new}^2 + I_{q,new}^2)} \tag{4.28}$$

The developed inverter model also includes specific limits for the inverter current, which in turn are used to define the limit for the output power according to the rating of the inverter. Therefore, the inverter model adjusts the power output (reducing the active power, while injecting reactive power) depending on the inverter rating. Three cases are developed in order to

maintain the injected current within the inverter rating conditions: Case 1 reduces the injected active current; Case 2 reduces the injected reactive current; and Case 3 reduces both active and reactive currents. The computations for limiting the injected current in the three cases is summarized in Algorithm 3. The inverter power output is maintained within inverter rating following the current limits. However, for activated fault ride-through operation, Case 1 is always applied to reduce the active current depending on the required reactive current at the point of common coupling to the grid.

Algorithm 3 : Limitation of inverter injected currents

Inputs: V_{inv}; $I_d = I_{d,new}$; $I_q = I_{q,new}$; I_{max}
if *Case 1* **then**
  if $|I_q| > I_{max}$ **then**
    set $I_d = 0$; $I_q = I_{max}$
    else
      set $I_q = I_{q,new}$, calculate I_d from Equation (4.28)
  else if *Case 2* **then**
    if $|I_d| > I_{max}$ **then**
      set $I_q = 0$; $I_d = I_{max}$
      else
        set $I_d = I_{d,new}$, calculate I_q from Equation (4.28)
  else if *Case 3* **then**
    calculate $I_d = I_q$ from Equation (4.28)

4.4 Simulation Results

Simulation results are presented in the current section to illustrate the integration of the proposed models in the transient stability analysis process using the time-domain simulation method described in Chapter 3. The main focus is to show the dynamic response and grid support capability of the new models. A network representation of the Baden-Württemberg transmission grid is used for the simulation tests in the current chapter. The structural representation of the modified Baden-Württemberg network is described and shown schematically in Appendix A.2. The network used for the actual simulation is defined in a Matpower case file format. In order to test the proposed models, the synchronous machines connected on buses 6, 8, 9 and 11 are replaced with PV and wind generators, i.e. PV on bus 6 and bus 8; and wind on bus 9 and bus 11. The connected generators are assumed to be equivalent representations of the aggregated generation in a PV solar park and wind-farm, since the purpose of the presented results is to illustrate the interaction of the renewable energy generators with the rest of the network. Two simulation cases are presented in the following subsections: The first case analyzes the grid support capability by comparing the response of the network to events (three-phase faults and load changes) with and without grid support functions for the renewable energy generators. In the second case, the change in system robustness following a network fault is assessed as the number of generators connected with grid support functions changes.

4.4.1 Analysis of Grid Support Capability

In the first part of the simulation, the proposed fault ride-through and reactive power support functions are tested. For this, a three-phase short circuit fault is simulated on bus 89 (cf. Figure A.2) at time $t = 1.0$ s and cleared after 150 ms. The PV and wind power generators are initially connected without network support functions, whereby they are controlled to disconnect from the network during the fault. Figure 4.6 shows the resulting bus voltage response. For purposes of illustration, only the voltages at the connecting points of the PV and wind power generators, the fault bus and a few selected buses are shown in the figure. As observed in Figure 4.6, there is a drop in voltage at the instant of fault occurrence. The voltage magnitudes of the corresponding buses at the beginning and end of the fault, respectively, in reference to the nominal voltage are summarized as follows: Bus6 – 67.8% and 67.4%; Bus8 – 66.7% and 66.3%; Bus9 – 85.2% and 84.9%; Bus11 – 85.5% and 85.1%.

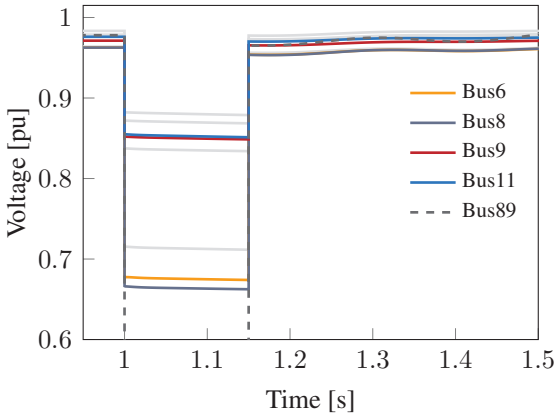


Figure 4.6: Bus voltage response for a fault duration of 150 ms without voltage support from PV and wind generator systems

Next, the PV and wind generators are controlled to remain connected during fault operation to support the network. For this, fault ride-through and voltage support functions are defined for the inverters of the individual generator. The following gain factors – k from Equation (4.23) – are defined for the respective generator: $k = 6$ and 2 for PV on bus 6 and bus 8, respectively; $k = 6$ and 2 for wind generator on bus 9 and bus 11, respectively. A similar three-phase fault is simulated as described in the first case. Figure 4.7 shows the voltage response at the buses of interest. There is a drop in voltage at all buses following the fault as observed in the previous simulation case, with similar voltage magnitudes at the beginning of the fault. However, different voltage magnitudes are observed at the end of the fault and are summarized as follows as a percentage of the nominal voltage: Bus6 – 73.6%; Bus8 – 68.7%; Bus9 – 86.6%; Bus11 – 86.0%. This shows that the enabled fault-ride through functions of the renewable generators result in an increase in bus voltage during the fault compared to the case without grid support functions. The increase in voltage is due to injection of reactive power by the inverter-connected generators to support the bus voltage.

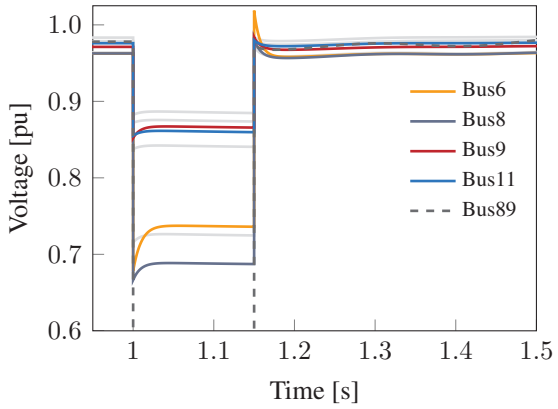


Figure 4.7: Bus voltage response during fault with grid support functionality

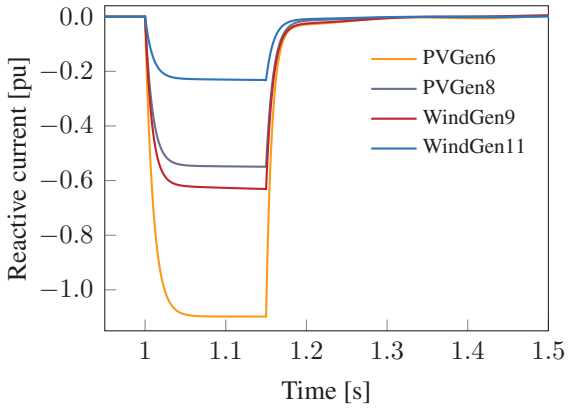


Figure 4.8: Injected reactive current for voltage support during fault

The injected reactive current is shown in Figure 4.8. The value of the injected reactive current during fault depends on the set gain factor k of each generator and the voltage drop, as defined in Equation (4.23). In this case, PVGen6 with $k = 6$ injects more reactive current than windGen9 with a similar k value, since the voltage drop on bus6 is more than that on bus9. Similarly, PVGen8 injects more current than windGen11, although both are set with a gain value $k = 2$. The amount of injected current in turn determines the level of voltage support at the respective connecting buses as shown in Figure 4.7. At the end of the fault period, the inverter-connected generators regulate their respective currents back to the initial value as observed in Figure 4.8, i.e. $I_q = 0$ in this case. It should be noted that the effect of voltage support is experienced on other buses as well. However, the impact is observed to depend on the bus location, where less impact is seen on buses located far from the inverter-connected generators providing voltage support. A detailed view of the voltage response on all buses is given in Appendix B.2.

Furthermore, the response of the generator side of the wind system following the fault is analyzed. The wind generator angular speed is analyzed for systems connected with and without a braking chopper in the DC-link. Figure 4.9 shows the response of generator angular speed. Following the fault without the braking chopper, the generators accelerate since less active power is injected into the grid compared to the pre-fault power. The generators experience a large speed deviation from the nominal operating point in the post-fault period. However, activating the braking chopper enables the generators to dissipate the excess power through the braking resistor, which results in a post-fault speed close to the pre-fault nominal operating point. The power dissipated in the braking chopper is depicted in Figure 4.10. From Figure 4.8, it is observed that WindGen9 injects more reactive current than WindGen11. This corresponds to a higher reduction in active power, resulting in WindGen9 dissipating more excess power in the braking chopper than WindGen11 as shown in Figure 4.10. Thus, the braking chopper enables the wind generator system to remain close to the nominal operating point following a fault, which is important for fault-ride through capability.

The next test case analyzes the active power reduction functions of the proposed inverter model. A network load change is simulated, where the total load active power is reduced by

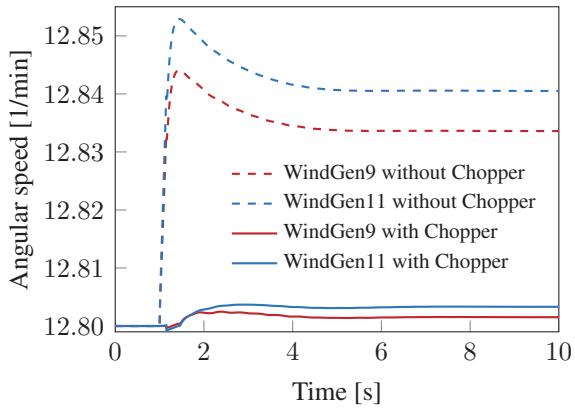


Figure 4.9: Response of generator angular speed to a network fault

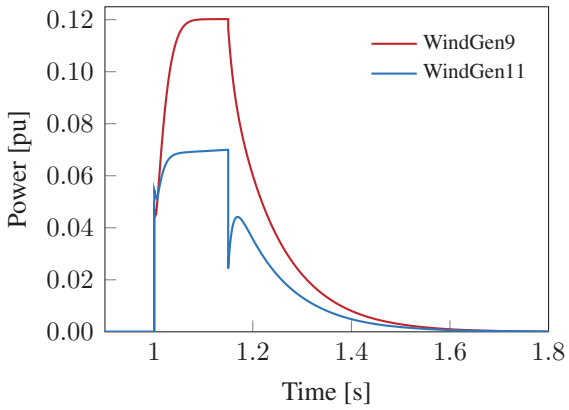


Figure 4.10: Dissipated power in braking chopper during fault ride-through

15% at $t = 1.0\,\mathrm{s}$. For illustration purposes, the power regulation by the synchronous machines is deactivated in the presented results. The resulting response in the system frequency is shown in Figure 4.11. As expected, a reduction in the load causes an increase in system frequency due to acceleration of the synchronous machines in response to the power imbalance. Without the active power reduction control function of the inverter, the new operating frequency is observed to be 50.78 Hz. However, activating the active power reduction function triggers the inverter-connected generators to reduce power injection when the system frequency exceeds a specified threshold value (50.2 Hz in the presented test case). The resulting reduction in active power of the inverter connected generators is shown in Figure 4.12. In this case, each generator reduces its active power injection by 18% of the value at 50.2 Hz. The frequency response with active power reduction is also depicted in Figure 4.11 and shows a new steady operating frequency of

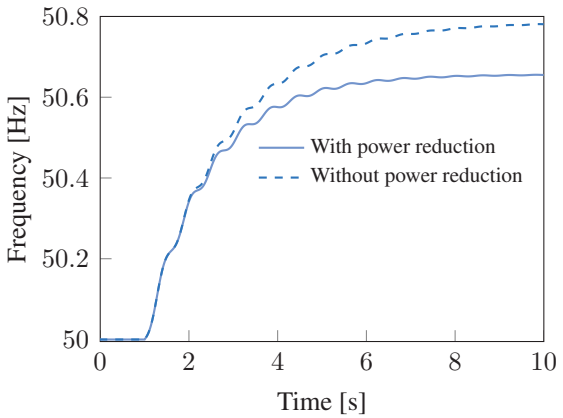


Figure 4.11: Frequency response to a load change with and without active power reduction by the inverter-connected generators

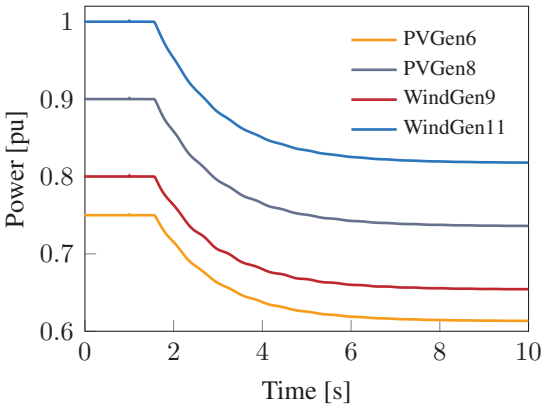


Figure 4.12: Active power reduction of the inverter-connected generators following a load change

50.65 Hz. This shows that the active power reduction function of the converters can potentially contribute to regulation of system frequency.

4.4.2 Assessment of System Robustness

The current test case analyzes the change in system robustness resulting from the replacement of conventional generators with renewable energies. Modifications in the generator setup are similar to the previous test case. The analysis is carried out by varying the number of renewable generators connected with grid supporting capability and compared to the original network consisting of only synchronous machines. Three-phase faults are simulated in each test case on bus 78 and bus 36 while varying the fault duration to determine the critical clearing time, which is used as a measure of the system's robustness. In addition, different gain factors k are applied to analyze the variation of system robustness with the control settings for voltage support during fault ride-through. The modified systems and the corresponding critical clearing times are summarized in Table 4.1.

Table 4.1: Assessment of system robustness using fault critical clearing time

Test System	System Description	Critical Clearing Time (ms)			
		Fault Bus 78	Fault Bus 36		
		$k = 4$	$k = 4$	$k = 2$	$k = 0$
System I	Network with only synchronous generators	632	660	660	660
System II	No FRT for the four inverter-interfaced generators	609	595	595	595
System III	FRT for PVGen8	613	607	610	611
System IV	FRT for PVGen8 and WindGen9	617	619	622	623
System V	FRT for PVGen6, PVGen8 and WindGen11	621	632	639	643
System VI	FRT for the four inverter-interfaced generators	625	644	652	657

As shown in Table 4.1, the original network with only conventional synchronous generators results in the highest critical clearing time of 632 ms for a fault on bus 78. The original network is therefore able to withstand the tested disturbance for a longer duration than the modified networks with integrated renewable energy generators. This behavior is attributed to the high system inertia provided by the synchronous machines, thereby reducing the rate of change of frequency (RoCoF) following the fault. The above behavior is further illustrated in Figure 4.13 showing the rate of angular separation of the critical generator from the rest of the generators following the fault in the different test scenarios. It is shown that the critical generator experiences a lower rate of angular separation in the original network (System I) compared to the modified networks. The trend in the modified networks shows that the critical clearing time increases with the number of renewable energy generators connected to the system with fault ride-through and grid support capabilities as shown in Table 4.1. In addition, the rate of angular separation of the critical generator also decreases with an increasing number of generators providing FRT and grid support during the fault as shown in Figure 4.13. A similar trend is observed for the additional fault case on bus 36 but with different critical clearing times, as expected, due to a change in fault location. This implies that there is potential improvement in

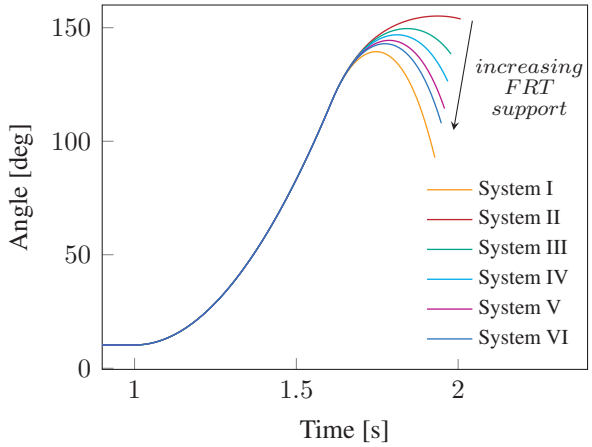


Figure 4.13: Change in rate of angular separation of the critical generator from the rest of the system with a varying number of inverter-connected generators providing FRT and voltage support

system robustness if renewable energy generators are controlled to provide grid support during disturbances. Furthermore, Table 4.1 shows that the CCT varies with the gain factor k. This is due to the fact that the gain factor k determines the required reactive power injection, which results in a reduction of active power injection by the inverter to meet the reactive power requirement. Therefore, optimal settings are necessary for the inverter control parameters in order to ensure a balance between injected active power and voltage support during FRT operation.

From the simulation results, it is shown that there is a significant reduction in system stability due to replacement of synchronous generators with converter-interfaced renewable energy generators. The reduction in stability can be attributed to the decrease in total system inertia. It is also observed that system stability can be improved if the interfacing converters are controlled to replicate the essential characteristics of synchronous machines to provide network support during disturbances. In general, the analysis shows that dynamic simulations can be used to derive the necessary adjustments in protection settings if synchronous generators are to be replaced by renewable energy sources.

4.5 Summary

The current chapter describes the modeling and integration of renewable energy sources in the time-domain simulation of power systems. The contributions in the chapter are the new dynamic models of solar PV and wind power generators as converter-interfaced generation systems. Development of the models is based on the notion that the dynamics of the converter perform the main role in the interaction of the renewable generators with the rest of power system. Thereby, the dynamic behavior of the generators during network transients is determined by the functional settings of the interfacing converters based on standard grid operation codes. Simulations have shown that renewable energy sources complying with network operator requirements can provide improved grid stability support. This is through the interfacing converters modeled to provide functions, such as voltage regulation via reactive power control, and frequency regula-

tion via active power reduction. Generally, the control functions of the interfacing converters provide renewable energy generation systems with features comparable to synchronous generators from a functional point of view. This chapter therefore addresses the open question requiring the dynamic modeling and integration of renewable energy sources in the transient stability analysis of power systems.

5 Proposed Parallelization of Time-domain Simulations

The previous chapters have focused on the verification of the accuracy of component models and computational methods developed for time-domain simulations. In addition to accuracy, another important factor in power system analysis is the performance of the computational methods. The previously described time-domain simulation algorithm is developed for sequential operation and optimized for running on single-processors, thereby rendering its application in large networks computationally challenging. Due to the ever increasing complexity in power system analysis, such a method is inadequate in terms of computational speed. Therefore, the current chapter describes a computational approach for time-domain simulations based on parallel computing in order to improve the computation speed in the transients stability analysis process. As illustrated in the computational results presented in Chapter 3, solving the algebraic network equation is identified to consume a huge amount of time in the time-domain numerical solution of the power system problem. In view of this challenge, the method in the current chapter proposes parallelization of the power system problem on the algorithmic level for the solution of the network algebraic equation. The parallel approach is based on restructuring of the network equation in such a way that it can be easily applied to formulation of a parallel algorithm. The parallel solution uses a parallel-in-space network decomposition and reformulation of the network equation coefficient matrix in a Block Bordered Diagonal Form (BBDF).

Grid partitioning is generally required for the spatial decomposition of the network in the parallel-in-space approach. For this, the current chapter further describes a new extended grid partitioning approach based on a multi-level graph partitioning technique. The extension in the grid partitioning approach is in such a way that an optimized partition is created consisting of boundary nodes which interconnect the main network partitions. Furthermore, simulation results are presented comparing the parallel method to the sequential algorithm presented in Chapter 3 in order to validate the level of accuracy and analyze the computational performance improvement in terms of speedup.

The current chapter is based on the work which has been published in [149, 150].

M. Kyesswa, P. Schmurr, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A New Julia-Based Parallel Time-Domain Simulation Algorithm for Analysis of Power System Dynamics," in *2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, Prague, Czech Republic, pp. 16–24, September 2020.

M. Kyesswa, A. Murray, P. Schmurr, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Impact of Grid Partitioning Algorithms on Combined Distributed AC Optimal Power Flow and Parallel Dynamic Power Grid Simulation," *IET Generation, Transmission & Distribution (In print)*, 2020.

5.1 Parallelization Methodology

The computational advancement in the presented method is achieved by a combination of the following factors: Using an efficient computing environment, applying an improved solution method, and application of a parallelization technique. To improve the performance using an efficient platform, the Matlab-based simulation algorithm described in Chapter 3 is extended to the Julia environment [113], which provides high-level and high-performance dynamic programming. In terms of the solution method, the developed solution of the power system problem is based on the decomposition of the network equation in a block bordered diagonal form (BBDF), and application of a set of efficient LU factorizations. With such a decomposition, the largest part of the set of system equations can be solved in a parallel scheme. In order to apply the parallel solution scheme in the presented method, an extended grid partitioning algorithm is developed to create parallelizable subnetworks. The current section describes the formulation of the parallelizable network solution and the grid partitioning methodology.

5.1.1 Formulation of Parallel Network Solution

The formulation of the parallel network solution proceeds from the general representation of the power system model given by Equations (2.36) and (2.37). From the structural analysis of the power system, it is observed that the generators and the connected controlling systems are naturally decoupled. This implies that the resulting differential and algebraic equation of the machines can be solved in parallel and only require local node variables to interact with the rest of the network. Defining a network consisting of p subsystems that can be combined to form the original unseparated network, the machine equations in each subsystem can be rewritten as

$$\begin{aligned} \dot{x_i} &= f(x_i, y_i, u_i) \\ 0 &= g(x_i, y_i, u_i) \end{aligned} \tag{5.1}$$

for $i = 1, 2, \ldots, p$, where x_i are the internal subsystem state variables, y_i includes the subsystem internal algebraic variables and the required local node variables to interact with the rest of the system, and u_i are the parameters of the i^{th} subsystem. The remaining coupled component of the algebraic equations is the network equation as represented in Equations (2.18) and (2.30).

To reformulate the network equation into a parallelizable form, the nodes in the nodal admittance matrix Y are reordered into a Block Bordered Diagonal Form (BBDF) [77, 78], with the p main partitions interconnected through the nodes in the $(p+1)^{th}$ partition. The network current balance equation in (2.18) written in BBDF structure is given by

$$\begin{bmatrix} Y_1 & & & & \bar{Y}_1 \\ & Y_2 & & & \bar{Y}_2 \\ & & \ddots & & \vdots \\ & & & Y_p & \bar{Y}_p \\ \bar{Y}_1^{\mathsf{T}} & \bar{Y}_2^{\mathsf{T}} & \ldots & \bar{Y}_p^{\mathsf{T}} & Y_s \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_p \\ V_s \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_p \\ I_s \end{bmatrix}. \tag{5.2}$$

In Equation (5.2), Y_i is the nodal admittance matrix of the internal i^{th} partition. If a partition contains n_i internal nodes, the dimension of Y_i is $n_i \times n_i$. The admittance matrix of the interconnect partition Y_s is an $n_s \times n_s$ matrix, where n_s is the number of boundary nodes in the interconnecting partition. $\bar{Y}_i$ is an $n_i \times n_s$ matrix containing elements of the branches connecting the internal nodes of partition i to the boundary nodes in the interconnect partition [76]. The node voltages and injected node currents are reordered accordingly, whereby I_i and V_i are the current injection and node voltage vectors, respectively, of order n_i in the i^{th} partition. The vectors I_s and V_s are the interconnect partition current injections and boundary node voltages, respectively, of order n_s.

The restructuring of the network problem in a form shown in Equation (5.2) allows the solution to be rearranged in p subnetworks that can be solved in parallel. In the presented method, the solution of Equation (5.2) is carried out in two steps. The first step constitutes a preprocessing sequential step for calculating the boundary node voltages V_s from the interconnect partition equation given by

$$\hat{Y}_s V_s = \hat{I}_s \tag{5.3}$$

where

$$\hat{Y}_s = Y_s - \sum_{i=1}^{p} \bar{Y}_i^{\mathsf{T}} Y_i^{-1} \bar{Y}_i \tag{5.4}$$

$$\hat{I}_s = I_s - \sum_{i=1}^{p} \bar{Y}_i^{\mathsf{T}} Y_i^{-1} I_i \tag{5.5}$$

Initially, matrix $\hat{Y}_s$ and vector $\hat{I}_s$ are explicitly formulated from the admittance matrices and current injections in the subsystems, respectively. The linear system in Equation (5.3) is solved using LU factorization for the voltages V_s. The boundary node voltages V_s are inserted into the respective subsystems, which enables the solution of the remaining blocks in Equation (5.2) to be separated into p sub-blocks that are solved in parallel in the second phase. The second phase is performed in parallel by applying LU factorization to solve for the node voltages V_i in the independent linear systems given as

$$Y_i V_i = I_i - \bar{Y}_i V_s \tag{5.6}$$

The solution process of the above network equation can be optimized by the following precomputations: Matrix $\hat{Y}_s$ in Equation (5.4) is calculated once at the start of the simulation loop and only recomputed at every event of change in network topology. Since the current vectors I_s and I_i change at every time step, vector $\hat{I}_s$ in Equation (5.5) needs to be calculated at every time step. However, the component $\bar{Y}_i^{\mathsf{T}} \cdot Y_i^{-1}$ in Equation (5.5) is precomputed at the start of the simulation. Furthermore, matrices $\hat{Y}_s$ and Y_i are factorized by LU decomposition before the main simulation loop; reformulation of the matrices is necessary only in case of occurrence of a network event, as described in Section 5.2.3.

The optimal performance of the solution algorithm requires a balanced distribution in the number of network nodes in each subnetwork. This is necessary for balancing the work load to the parallel processors. Moreover, it is shown that the solution formulation using the block bordered diagonal form initially requires a sequential step to solve for the boundary node voltages.

The complexity of such a computation grows with more than linear complexity. Therefore, in order to reduce the sequential computation time, the number of boundary nodes in the interconnect partition should be minimized in relation to the dimensions of the network. In the presented method, the number of boundary nodes is optimized by minimizing the number of cut branches in between partitions, which is obtained using an efficient grid partitioning strategy. The next section describes the extended partitioning strategy for application to dynamic simulations.

5.1.2 Network Partitioning

Power grids can be naturally represented as graphs. The requirements highlighted in the previous section for optimizing the parallel solution of Equation (5.2) are similar to the requirements in graph partitioning [151], where the objective is to minimize the number of cut edges between graph partitions. Thereby, network partitioning can be formulated as a graph partitioning optimization problem. A detailed overview of the advances in graph partitioning and existing algorithms is given in [152, 153, 151], of which the approaches used for obtaining good partitions in large graphs are based on multilevel graph partitioning [154].

The power grids in the present thesis are considered to be unweighted graphs, whereby an equal weight function of one is set for every branch. In this case, a minimum number of cut edges results in a minimal number of branches between the subnetworks. The current section describes the grid partitioning strategy applied for the spatial decomposition of the grid into parallelizable subsystems. In the first step, a multilevel graph partitioning approach is applied to create equally sized partitions with a minimum number of cut branches. This format is referred to as the *basic partitioning format* in the rest of the present thesis. However, the parallel solution formulated in this work requires an interconnect partition for information exchange between subnetworks. Therefore, an extension of the graph partitioning strategy is developed to create the interconnect partition consisting of boundary nodes. This extended partitioning format is referred to as the *interconnect partitioning format* in the rest of the current thesis.

Graph Partitioning: Basic Partitioning Format

The multilevel graph partitioning approach applied for the basic partitioning format is the Karlsruhe Fast Flow Partitioner (KaFFPa) [154, 155]. In general, a multilevel graph partitioning algorithm creates equally sized partitions with a minimum number of cut edges in a three-step process: coarsening, initial partitioning, and refinement phase. In the coarsening phase, the algorithm contracts the input graph into a smaller graph that can be partitioned with global algorithms. The graph contraction is based on a matching strategy identifying an independent set of edges with no common vertices [156]. A global path algorithm [156] is used as a matching algorithm to create a maximum matching in order to determine the edges to contract. A coarse graph is then formed by combining the vertices in the matching set and updating the weights of each new vertex as the sum of the weights of the previous vertices; thus decreasing the size of the input graph. At the initial partitioning stage, KaFFPa uses the SCOTCH algorithm [157] for global partitioning if the graph size is small enough. The formed contractions are then iteratively undone at the refinement phase using local search based refinement algorithms to systematically move vertices between partitions in order to reduce cut edges, while maintaining the balancing

constraints. A detailed description of the KaFFPa algorithm is given in [154, 155]. This graph partitioning results in evenly sized partitions with minimal cut edges. The output of the basic partitioning format is a graph with p partitions and stored in form of a vector, where the n^{th} row contains the partition index in which node n is located. A comparison of the partitioning results with other graph partitioning and spectral clustering algorithms is given in [150, 158].

Extension of Graph Partitioning: Interconnect Partitioning Format

Considering that the restructured coefficient matrix in Equation (5.2) consists of a set of boundary nodes forming an extra partition (interconnect partition), in addition to the p main partitions, the output of the basic partitioning cannot be directly applied for the parallel solution. The interconnect partitioning format is developed to derive suitable partitions under the condition that the interconnection between the main partitions is through the boundary nodes in the interconnect partition. Therefore, the interconnect partition is created by moving adjacent boundary nodes from the p partitions into a $(p+1)^{th}$ partition, which eliminates any direct connection between the different partitions.

The following procedure summarizes the interconnect partitioning format process: As a first step, nodes are ranked according to the number of edges connecting to other partitions. The node with the most number of edges to other partition is ranked highest and is moved to the interconnect partition. The ranking of the nodes is updated, with the unmoved neighboring nodes becoming eligible for selection and the process is continued. However, if multiple nodes have the same rank, the secondary selection criterion is the size of the main partitions. Selection priority is given to the node from the largest partition. This selection criterion minimizes the size of the interconnect partition for a given set of input partitions and balances the partitions in terms of partition sizes. The partitioning process is terminated when all the boundary nodes connecting to other partitions are eliminated and the connection between partitions exists only through the interconnect partition. For an output of the basic partitioning format consisting of p partitions, the interconnect partitioning format results in $p+1$ partitions.

Figure 5.1 illustrates the transformation from the basic partitioning format to the interconnect partitioning format. The arrows in the figure indicate the data streams for the interaction between partitions, which shows that the nodes in the interconnect partition provide the link for information exchange between the main partitions. As shown in the simple system with three main partitions in Figure 5.1, the unknown boundary node voltages in the interconnect partition are calculated sequentially and sent to the main partitions. The complexity of the sequential solution therefore depends on the size of the interconnect partition, which is optimized by minimizing the number of cut branches between partitions from the basic partitioning format. The system of equations in the main partitions are solved separately in parallel.

5.2 Computation Process

The computation process in the presented parallel dynamic simulation method is summarized in the following three main steps: initialization, solution of system trajectory and event handling.

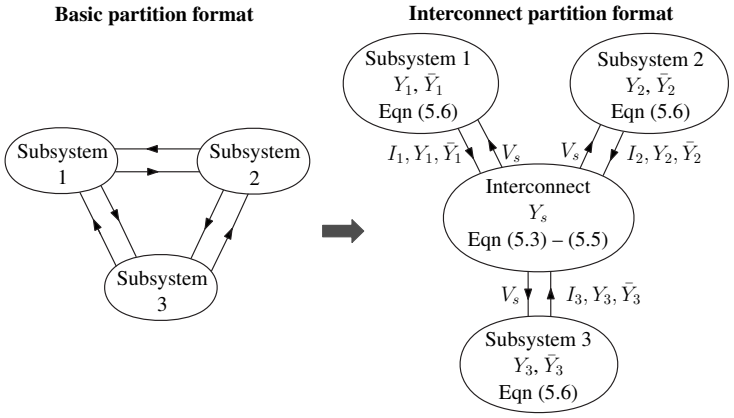


Figure 5.1: Interconnect-partitioning format for a three-subsystem network

An overview of the workflow in parallel dynamic simulation method is shown in Appendix B.3. The current section describes the main steps of the computation process.

5.2.1 Initialization

The initialization step is required at the beginning of the dynamics simulation to initialize the system to steady state. At this step, the objective is to determine the initial values of the dynamic state variables x_0 and algebraic variables y_0 required at the start of the numerical computation for the actual system trajectory. The variables can be either computed or specified according to known initial network operating conditions. In the presented method, steady state power flow computations are used to define the initial operating conditions. The power flow calculation is carried out on the original unpartitioned network to obtain the initial bus voltage magnitude and phase angle, and the active and reactive power at the network nodes.

With the assumption that the system is operating in steady state equilibrium prior to the start of the simulation, all time derivatives in the equations describing the machines and connected controllers are set to zero at $t = 0$. From Equation (5.1), the resulting system of equations at $t = 0$ is given by Equation (5.7). Solving Equation (5.7) for x and y gives the initial dynamic state variables x_0 and algebraic variables y_0.

$$\begin{aligned} \dot{x} = 0 &= f(x, y) \\ 0 &= g(x, y) \end{aligned} \tag{5.7}$$

5.2.2 System Trajectory

The system trajectory is obtained from the solution of the system of differential and algebraic equations (2.36) and (2.37). The differential equation (2.36) is solved numerically using numerical integration methods which approximate the solution of x at t_{n+1} from a sequence of previously estimated values at t_n, t_{n-1}, ..., t_0. Generally, an integration method advances the solution in each time step from $t = t_n$ to $t = t_{n+1}$, with a step size $\Delta t = t_{n+1} - t_n$. The approximate solution of the state variables x_{n+1} at the current time step t_{n+1} is given in terms of the

step size Δt_n, the previously computed values x_n, x_{n-1} ..., and the functions $f(x_{n+1}, y_{n+1})$, $f(x_n, y_n)$, $f(x_{n-1}, y_{n-1})$,..., depending on the discretization method. In the parallel computation method presented in the current chapter, the numerical integration is based on an explicit, one-step method, which approximates the solution at each time step based only on the computed values from the immediately previous time step t_n. This is the fourth order Runge-Kutta method [128]. However, the additional numerical solvers described in [127] can also be applied in this computational method. With the explicit Runge-Kutta method, the general solution of the differential equation at time t_{n+1} is given in the form of Equation (5.8). At the point t_{n+1}, the algebraic equation is then expressed in the form shown in Equation (5.9).

$$x_{n+1} = x_n + f(x_n, y_n) \tag{5.8}$$

$$0 = g(x_{n+1}, y_{n+1}) \tag{5.9}$$

An alternating solution scheme [129] is applied in the presented method in order to solve the system of equations in (5.8) and (5.9). In this scheme, the discretized differential equations and algebraic equations are solved separately at every integration step. The set of discretized system of Equations in (5.8) is solved for x_{n+1}, which is then substituted into Equation (5.9) to solve for y_{n+1}. It should be noted that the alternating solution scheme is susceptible to interface errors, which arise as a result of solving equations (5.8) and (5.9) alternately for variables x and y. To obtain solutions with less interface errors in the presented computation method, the algebraic equations are solved at every integration stage. This implies that the algebraic equations are solved four times at each simulation time step, corresponding to the four stages for computing the increments approximating the derivative in the fourth order Runge-Kutta method applied in the presented method.

The network equation is the main part of the algebraic equations. Similar assumptions highlighted in Section 3.2.3 are considered to simplify the network equation into a linear algebraic equation that can be solved efficiently using the LU factorization method. The solution of the network equations in the sequential algorithm considers the original unpartitioned network and applies a single LU factorization to decompose the entire network admittance matrix. In contrast, the parallel solution uses multiple LU factorizations on the subnetwork admittance matrices, of which the individual partition network equations are solved in parallel as described in Section 5.1.1. In both sequential and parallel solutions, the network equation is considered to be linear. The admittance matrix is therefore only affected by network switching events. Thereby, the L and U factorization components of the admittance matrix are reused until the next switching event occurs. With this simplification, the network solution requires only the forward and backward substitution for the largest part of the simulation.

It is important to note that the computational speed and accuracy of the system trajectory also depend on the integration step size. Therefore, care must be taken when selecting the step size bearing in mind that the DAEs in power systems are known to be stiff in nature, consisting of a wide range of time varying dynamics. The explicit fourth order Runge-Kutta integration method applied in the presented method requires fixed small step sizes for significant simulation accuracy and to avoid numerical instability. The maximum integration step size is defined by the smallest time constant in the system model.

5.2.3 Event Handling

The third step of the computation process is the event handling step, which represents the transient behavior of the power system in the presence of disturbances. The disturbances can be in form of event disturbances or load changes. Event disturbances include loss of generation, short circuit faults, sudden load changes, which result in a change in system configurations. On the other hand, load disturbances are caused by the continuous small variation in system loads, without changing the system configuration. By dividing the system under analysis into pre-disturbance, disturbance and post- disturbance sections, the event handling function computes and defines the transition from one continuous section to another. The events considered in the presented computational method are limited to time events, which are predefined to occur at a specific time during the simulation. The events are defined using a time vector describing the start and end time, nature of event, affected network parameters and location.

During the simulation, an event is detected by referring to the defined event time vector. If the set start-time is reached, the event type, location, and affected network parameter are loaded. In the disturbance section, events result in changes in system equations. For example, a short circuit fault is simulated by inserting a high shunt admittance on the bus where the fault is located. Thereby, the shunt admittances of the network are modified during the fault period. Other simulated events include load changes, where the active and reactive power at the buses are varied, and change in line parameters, in which the line resistance and reactance of a branch are changed.

For the sequential algorithm, the whole admittance matrix is recomputed and refactored using the LU factorization method. Since the network is structured into partitions in the parallel algorithm, the only admittance matrix that changes is the one where the fault is located. The same procedure for construction and factorization of partition admittance matrices described in Section 5.1.1 is repeated. The network equations are then solved for the bus voltages at the time immediately following the event. The system continuous trajectory in the fault section is obtained by numerical integration using the new solution of algebraic variables. Clearing the disturbance is regarded as another event and a similar procedure is carried at the set time for the end of the event. At this point, the new network parameters defined in the event time vector are loaded. If the fault is cleared by resetting the shunt admittance to its original value, the resulting post-disturbance network structure is similar to the pre-disturbance network. Otherwise, setting a different value results in a different network structure for the post-disturbance network.

Depending on the simulation step size, special care should be taken when setting the event time vector so that the fault time can be captured by the step size. To ensure that the right point of the event time is captured, a tolerance value ε is allowed between the simulation time t and event time t_{event} given by $t - t_{event} \leq \varepsilon$. The value ε is set small enough to capture events at the right time $t = t_{event}$, and large enough not to skip events.

5.3 Implementation Aspects

The current section describes the implementation aspects in the presented computational method. Firstly, the implementation of the extended graph partitioning approach is described, including

the special functions for the formation of the required input network files, in addition to the graph partitioning program. This is followed by the implementation of the parallel computational method in the Julia programming environment. Finally, the communication aspects in the computational process are described in relation to the applied Julia programming environment.

5.3.1 Interconnect Partitioning Algorithm

The partitioning process starts with the definition of the original unpartitioned network in a *Matpower* casefile format. The input casefile contains the topological information required for partitioning the network into subnetworks. Initially, the network topology is converted into a graph format which can be input into a graph partitioning program. In the partitioning approach developed in the present thesis, KaFFPa is used as the supporting partitioning program. KaFFPa uses the Metis graph format [159] to redefine the network topology and requires undirected graphs, without self-loops and parallel edges. Therefore, the presented partitioning approach initially transforms the network into a Metis graph format in which a graph is stored in a plain text file with $n + 1$ rows, where n corresponds to the number of vertices. The first row contains the number of vertices, the number of edges, and an entry for the associated vertex and edge weights. Since the input networks in the present thesis are considered to be unweighted, the third entry on the first row is therefore ignored. The following n rows in the file describe the information of each vertex, with indices of the vertices connected to the current row vertex written in text format.

The input to the partitioning program – KaFFPa – is the Metis graph text format, and the required number of graph partitions p. The program outputs the partitioned graph in text format containing n rows, where the n^{th} row corresponds to the n^{th} vertex of the graph. Each row gives information about the partition (1 to p) in which the corresponding vertex is located. This output defines the basic partition format. A Matlab algorithm is developed to convert the basic partition format to the interconnect partition format. The general procedure for conversion the algorithm is based on the partitioning process described in Section 5.1.2. The output of the function is a partitioning file in text format with $p + 1$ subsystems that is used as the input for the parallel dynamic simulation algorithm. The network partitioning process is summarized in Algorithm 4 and Algorithm 5.

Algorithm 4 : Basic partitioning format

Inputs: Network casefile; $n-$nodes, $b-$branches
Required partitions p
Eliminate recycling branches
Build graph G in text format from network topology
Partition G into p subsystems
Return: Graph G and p sub systems

Algorithm 5 : Interconnect partitioning format

Inputs: b branches, partition indices $(1 \dots p)$ for nodes
Assign interconnect partition index $p + 1$
for *each branch* **do**
  Determine partition indices of $from$ and to nodes
  if $from$ *partition* $\neq$ to *partition* $\&$ *partition index* $\neq p + 1$ **then**
    branch $\rightarrow$ cut branch; nodes $\rightarrow$ boundary nodes
  Ranking boundary nodes:
  for *boundary nodes* **do**
    Rank based on branch count to other partitions
    if *nodes exist with equal branches* **then**
      Rank based on partition size of node location
    Move highest ranking node to index $p + 1$
    Update list of boundary nodes
    Repeat Until set of boundary nodes is empty
Return: partitions 1 to p and interconnect partition $p + 1$

5.3.2 Parallel Dynamic Simulation Algorithm

The network partitioning described in Section 5.3.1 is a preprocessing step that provides inputs for the parallel dynamic simulation, in addition to the original unpartitioned network defined in *Matpower* casefile format. The power flow at the initialization stage of the parallel dynamic simulation is computed using the *PowerModels* package [160] in Julia. The network data format required by the *PowerModels* package is consistent with the *Matpower* file format, which allows the network casefiles defined in *Matpower* to be directly used in the Julia implementation. In addition, the package *Matlab.jl* provides an interface for using Matlab functions in the Julia environment. With this package, the basic input functions defined in the Matlab time-domain simulation, such as network event files and dynamic model parameters, are directly called within the Julia implementation of the algorithm.

The admittance matrices of the original unpartitioned network and each partition are built from the original network structure and the subnetwork graph data, respectively. Additionally, initial variables from the power flow solution are required for the estimation of the constant load admittances. The matrices at this stage define the network in a steady state condition.

In the presented method, the parallelization of the computations is limited to a single time step since the dynamic simulation process is based on a step-by-step numerical solution. There are two main steps computed in parallel in a single time step: the solution of the decoupled machine differential equations, and the BBDF based network solution. The solution of the machine differential equations to obtain the current injections is based on the fourth order Runge-Kutta method as described in Section 5.2.2. The solution of the network equations consists of the precomputation steps, which are mainly matrix construction steps required for the sequential solution of the interconnect partition equations, and the parallel solution of the subnetwork equations. For the task of solving the linear network equation in each partition, an efficient

Algorithm 6 : Main steps in the parallel computation method

Inputs: Network casefile and partitions
Initialization: V_0, X_0
Precomputation:
  Form subsystem matrices Y_i, $\bar{Y}_i$ and Y_s
  Compute $\hat{Y}_s$ and product $\bar{Y}_i^{\mathsf{T}} \cdot Y_i^{-1}$
  LU factorize Y_i and $\hat{Y}_s$
for *each partition* **do**
    Calculate machine state variables X_i
    Compute current injection in each partition I_i
Compute link currents $\hat{I}_s$
Solve for the interconnect subnetwork voltages V_s
for *each partition* **do**
    Solve for subnetwork node voltages V_i
Return: State and algebraic variables at each time step

sparse LU factorization solver is applied from the UMFPACK library [161]. The main steps of the parallel computation method are summarized in Algorithm 6.

5.3.3 Communication Aspects

From the above description of the parallel dynamic simulation, it is observed that the key computations in the solution process are memory bound tasks dealing with vector arithmetic, matrix multiplication, and solving the network equation. Such a parallelization problem can be effectively handled in the Julia programming environment using multithreading constructs according to the analysis detailed in [162].

Figure 5.2 depicts the simulation time line for the parallel dynamic simulation for a system with two main partitions (p_1, p_2) and an interconnect partition. The variables in the figure represent the system variables solved at the corresponding stages in each partition. The communication aspects during the parallel computation are summarized as follows: The initial step is carried out on the main processor using the unpartitioned network to establish the quasi steady state conditions of the system. The steady state algebraic variables derived from the unpartitioned network conditions are sent to the respective subnetworks. At this point, the subnetworks begin to compute the corresponding initial dynamic state variables and precompute the internal admittance matrices and the boundary matrix elements as shown in Figure 5.1. Additionally, the precomputation stage in each partition also includes the LU factorization of the internal subsystem admittance matrices required for computing the node voltages in Equation (5.6). After the precomputation step, the interconnect partition receives the admittance matrices from each partition and starts to precompute the interconnect admittance matrix from Equation (5.4).

Assuming the main simulation starts at time t_0, the subnetworks start solving the assigned machine differential equations to compute the state variables and the resulting node current injections. The interconnect partition then starts receiving the current injections from the sub-

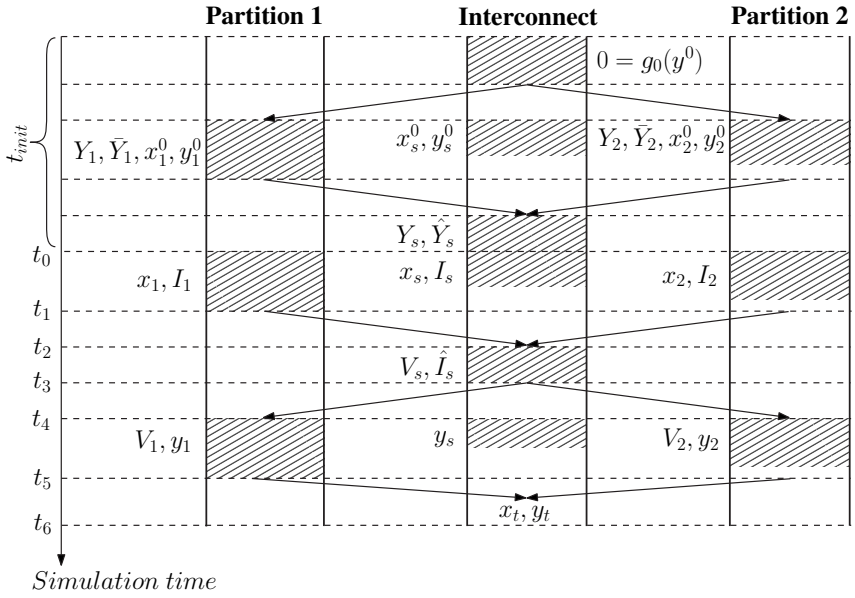


Figure 5.2: Representation of communication aspects for a two-partition system showing the initialization step (t_{init}) and one simulation time step ($t_0 - t_6$)

networks at time t_1 to compute the link currents according to Equation (5.5) starting at t_2. The boundary node voltages are then computed from Equation (5.3). At t_3, the subnetworks start receiving the boundary node voltages from the interconnect partition. Each subnetwork then separately solves Equation (5.6) for the node voltages starting at t_4. Once each subnetwork has finalized updating the node voltages at t_5, the local dynamic state variables and algebraic variables are sent to and saved by the main processor at t_6.

The execution time for one simulation time step is the duration from t_0 to t_6. This is true for a simulation with a fixed network topology. However, any change in network operating conditions causes a change in topology, which necessitates reformulation of the corresponding admittance matrices in the subnetworks as described in the event handling process. A time step involving event handling therefore additionally includes the computation time for reformulating and factorizing the admittance matrices, and the solution of the algebraic equations (5.3) to (5.6) to update all algebraic variables before the next time step.

5.4 Validation and Performance Evaluation

5.4.1 Simulation Setup

The results presented in the current section are obtained using the high performance computing cluster ForHLR II at Karlsruhe Institute of Technology (KIT). The computing cluster consists of multiple nodes, each with 20 usable cores, two Intel Xeon E5-2660 v3 Deca-Core processors and 64 GByte of RAM. Interconnection between the nodes is through a high speed InfiniBand 4X EDR Interconnect link. Further details about the ForHRL II computing cluster are given in

[163]. The results presented in this section are performed on a single computing node of the cluster.

Standard test networks, as described in Chapter 3 (cf. Table 3.1), are applied for validating the computational algorithm. In each test case, a simulation period of $10\,\mathrm{s}$ is considered, with a step size of $1\,\mathrm{ms}$. Additionally, two events are simulated in each case. The first event is the onset of a short circuit fault on a bus applied by changing the shunt value on a specified bus to a high value for a defined period of time. The second event corresponds to clearing the fault by resetting the bus shunt value to its original value. This creates an additional complexity in the computation, since the network admittance matrix does not remain constant throughout the simulation.

The assessment in the current section is divided into two parts. The first part focuses on the validation of accuracy in the presented parallel computational method. In the second part, the performance of the parallel method is evaluated in terms of computational speedup, in comparison to the fastest sequential method. Different test network structures are used for the performance evaluation in order to assess the speedup with varying network sizes and number of network partitions.

5.4.2 Validation of the Proposed Parallel Solution

In the first part of the evaluation, the accuracy of the presented parallel dynamic simulation method is validated against the sequential method described in Chapter 3. The simulation results in the two algorithms are compared to evaluate the level of accuracy of the component models and numerical solution strategy in the presented parallel computational method. For illustration purposes, the IEEE 30-bus system (Case30 in Table 3.1) is used as the test network in the validation process presented in this part. The structural representation of the network and data used for setting up the network are given in Appendix A.3. The applied generator and controller models are similar to those used in the test case described in Section 3.3.1. Using the new extended grid partitioning scheme, the network is partitioned into three subsystems as shown in Appendix A.3, where the resulting final partitioning information of the network is summarized in Table 5.1.

Table 5.1: Summary of network partitioning for IEEE 30-Bus network

Partition	Bus index	Bus count
Partition 1	1, 2, 3, 5, 7, 8, 9, 11	8
Partition 2	21, 22, 24, 25, 26, 27, 29, 30	8
Partition 3	12, 13, 14, 15, 16, 17, 18, 19, 20	9
Interconnect	4, 6, 10, 23, 28	5

In the above partitioning files, the numbers represent the bus indices in each partition. The simulation scenario considered in this part of the analysis is a three-phase to ground fault on bus 18 applied at time $t = 1.2\,\mathrm{s}$ for a duration of $50\,\mathrm{ms}$. The simulation is run for $10\,\mathrm{s}$.

Figure 5.3 to Figure 5.5 show the comparison of the simulation results obtained using the presented parallel method (*Par*) in Julia to those from the sequential method (*Seq*) in Matlab.

The variables used for the comparison are the generator relative rotor angle, the rotational speed deviation, and the magnitude of the bus voltages. In Figure 5.3, the generator on bus 1 is assumed to be the reference machine. Therefore, the relative rotor angles are computed with respect to angle δ_1 of generator 1, i.e. Gen2: $\delta_{2,1} = \delta_2 - \delta_1$, Gen22: $\delta_{22,1} = \delta_{22} - \delta_1$ and Gen27: $\delta_{27,1} = \delta_{27} - \delta_1$. Key generator buses are selected for illustration of the voltage response in Figure 5.5. Further details of the rotor angle, rotational speed, and voltage magnitude at generator buses and fault bus are given in Appendix B.3.2.

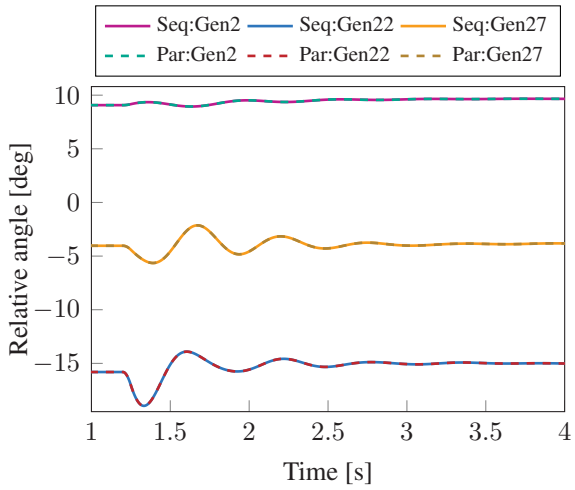


Figure 5.3: Comparison of generator relative rotor angles following a bus fault in the sequential Matlab-method and the proposed parallel dynamic computational method

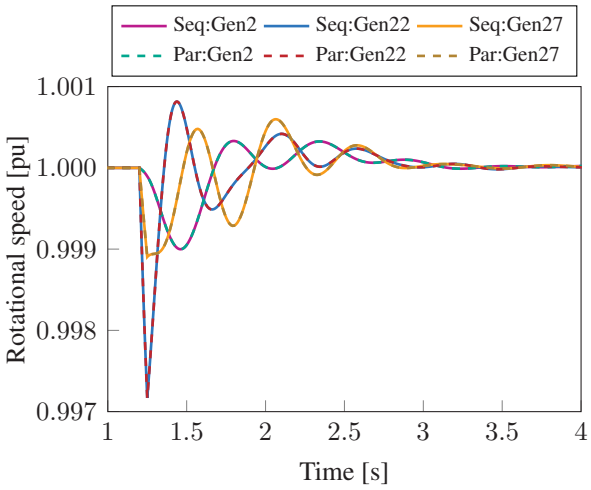


Figure 5.4: Comparison of generator rotational speed following a bus fault in the sequential Matlab-method and the proposed parallel dynamic computational method

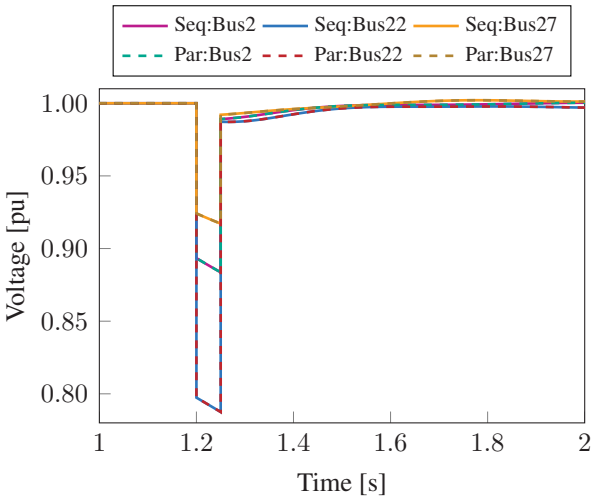


Figure 5.5: Comparison of bus voltage response following a fault in the sequential Matlab-method and the proposed parallel dynamic computational method

The simulation results in Figure 5.3 to Figure 5.5 show a perfect match in the accuracy between the proposed parallel computational method and the sequential Matlab-based method. Such a similarity in the results shows that the component models and the numerical solution strategy implemented in the presented parallel method match those in the validated sequential algorithm. This implies that the BBDF formulation of the network equation into subnetworks correctly replicates the results of the original unpartitioned network equation formulation. Regarding the solution strategy, the similarity in the simulation results can be attributed to the fact that both algorithms apply a similar numerical solution approach based on the fourth order Runge-Kutta method and LU factorization for the linear algebraic equations.

5.4.3 Performance Evaluation

The second part of the evaluation assesses the performance improvement of the proposed parallel method in terms of computation speedup in reference to the sequential method. The test networks applied for the performance evaluation represent standard network systems of varying structural complexity as shown in Table 3.1. The networks are partitioned into different sets of optimized partitions (subsystems) using the new extended interconnect partitioning approach. Each set of partitioning is applied to the parallel computational method to determine the number of partitions (partition count) resulting in the optimal simulation runtime. Table 5.2 summaries the optimal partitioning count for each network resulting in the best runtime in the parallel method. Details of the partitioning information and the corresponding parallel simulation runtime are given in Appendix B.3.3.

Table 5.2: Optimal partitioning count from the new extended interconnect partitioning approach

Network	Number of buses	Partition count	Average size	Interconnect size	Partition difference
Case9	9	2	3	2	1
Case30	30	4	6	6	2
Case118	118	6	16	18	1
Case300	300	5	57	15	2
Case1354	1354	7	188	37	5
Case9241	9241	16	567	161	41
Case13659	13659	10	1356	97	85

In order to analyze the performance improvements, the parallel method is compared to the Matlab-based method and its corresponding sequential extension in the Julia environment. Figure 5.6 depicts a graphical representation of the minimum computational runtimes in the three methods considering the optimal partitioning count for each network. From the results shown in Figure 5.6, it can be observed that both the sequential extension and new parallel method in the Julia environment show a great performance improvement over the sequential Matlab-based method. This performance improvement is attributed to the high performance capability provided by the Julia programming environment.

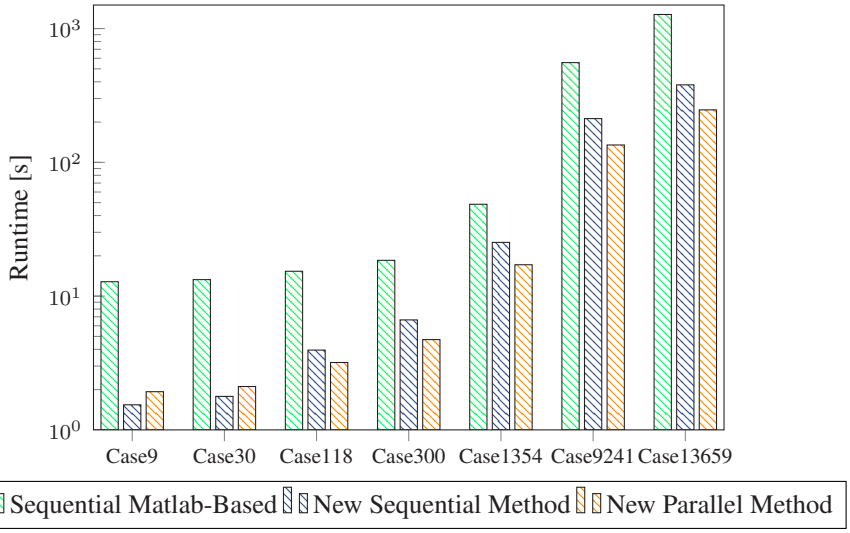


Figure 5.6: Comparison of computational runtime of dynamic simulations in the sequential Matlab-based method, new sequential Julia-based method, and the new parallel Julia-based method

With the above noted performance improvement in mind, the extended sequential method in Julia is used for further evaluation of the attained speedup of the parallel computation algorithm. The speedup is computed according to the relation $Speedup = T_s/T_p$, where T_s is the execution time in the sequential method and T_p is the runtime in the parallel method of the respective test cases. Figure 5.7 shows the simulation speedup of all the possible network partitioning counts for each network. In the presented evaluation, the largest partitioning count is limited to 20 partitions, which corresponds to the core limit of the single computing node of the used ForHRL II computing infrastructure. A partitioning count equal to one represents the simulation using the sequential extension of the computation method in Julia.

Analyzing the computation speedup in Figure 5.7 shows that the parallel simulation runs relatively slower than the sequential simulation for all partitioning counts in cases with small networks. This slowdown in computational performances can be attributed to the fact that the increase in complexity of the parallel solution structure using the block bordered diagonal formulation and communication overhead between the processes outweigh the parallelization benefits. However, the parallel computation shows significant improvements in speedup with increasing network sizes. Figure 5.8 illustrates the speedup achieved for the various partitioning counts in the larger networks. The reference speedup of 1.0 is the point at which the computational runtime of the parallel algorithm is equal to that of the sequential algorithm. The best speedup for the three largest networks considered in the current evaluation is summarized as follows: Case1354 – 7 partitions – 46.96% speedup; Case9241 – 16 partitions – 57.46% speedup; Case13659 – 10 partitions – 53.8% speedup. Therefore, the significant speedup implies that the gain in performance due to the parallel solution with the larger networks is greater than the communication overhead between processes and introduced parallel complexity.

An important observation from the presented results is that the parallel simulation speedup varies with the number of network partitions. This proves that an optimal partitioning count

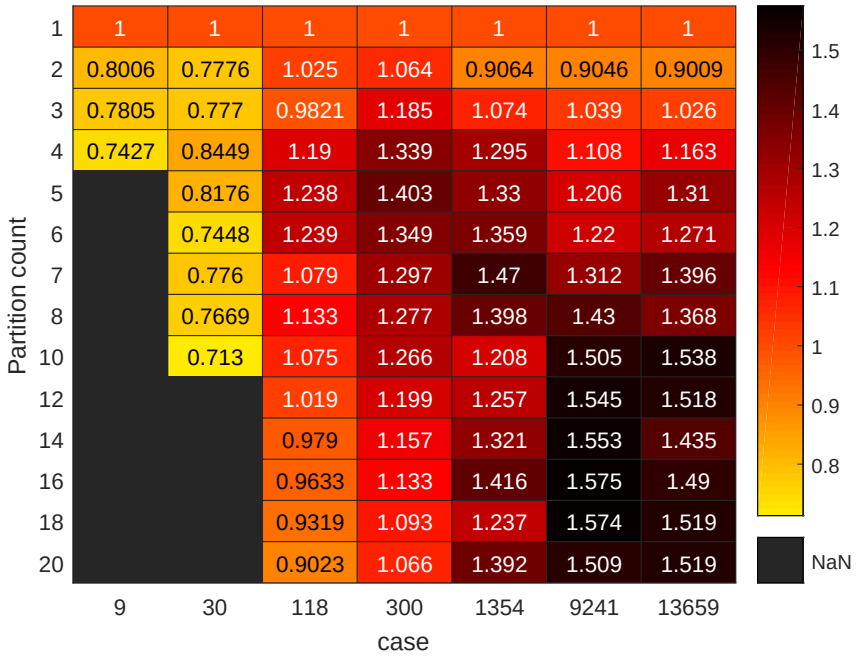


Figure 5.7: Computational speedup for each partitioning count of the corresponding test network cases

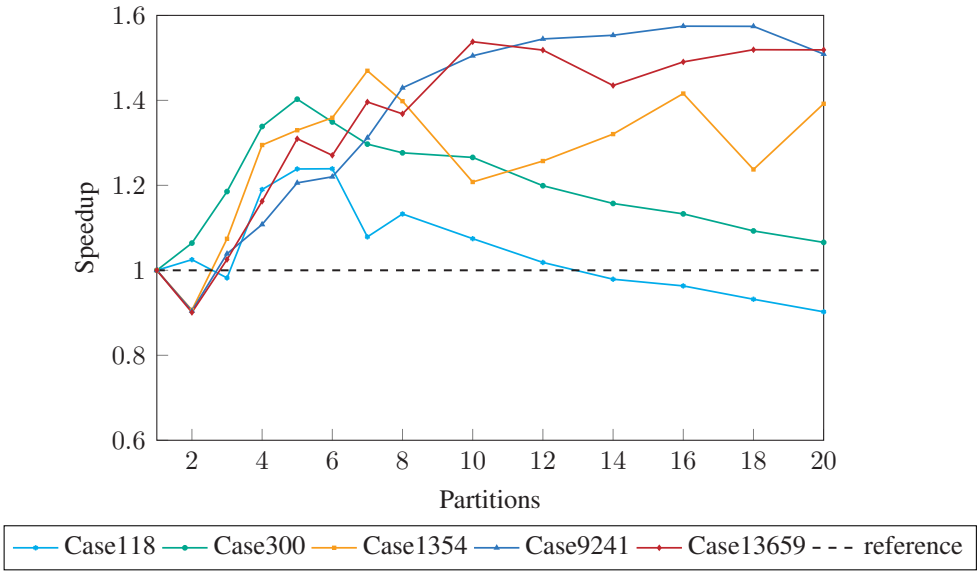


Figure 5.8: Summary of computational speedup for different partitioning counts in the large test networks

exists for each network, which results in the best balance between the partition sizes in order to create balanced parallelizable tasks and the interconnect partition size to minimize the sequential task during the computation process. Therefore, the runtime of the parallel dynamic simulation depends on the optimal partitioning of the network. A high number of partitions results in a large interconnect partition, which increases the sequential runtime of the network

solution. At the same time, the parallelizable partitions differ and decrease in size, causing an increase in waiting times and less parallelizable tasks.

5.5 Summary

The current chapter addresses the open question regarding the need for improved methods in order to address the increasing computational complexity in transient stability analysis. The main contributions in this chapter are summarized as follows: A new extended graph partitioning approach used for defining parallelizable subsystems in dynamic simulations; and a parallel dynamic computational method using a high performance programming environment, an improved solution method and high performance computing infrastructure. In terms of accuracy, the sequential and parallel Julia algorithms show a similar level of accuracy in deriving the system state of stability and actual system trajectory compared to the validated sequential Matlab-based algorithm. The new computational method benefits from the performance improvement provided by the Julia environment, resulting in significant runtime speedup compared to the Matlab-based simulation algorithm. According to the presented simulation results, the parallel-in-space scheme applied in the formulation of the network solution achieves significant speedup with increasing network sizes. In addition, the quality of network partitioning affects the runtime in the parallel method, since the speedup varies with the number of partitioned subsystems and reaches the highest point for an optimal partitioning count.

6 Fast Dynamic Stability Assessment Method

The time-domain simulations presented in the previous chapters have shown that the intensive step-by-step numerical integration in the solution process results in a huge computational burden in system analysis. Even with the development of parallel solutions, which can benefit from advancements in computing hardware and in high performance computing techniques, applying such methods in dynamic security assessment is generally a time consuming process. This is seen from the analysis perspective, which necessitates consideration of a large number of contingencies for the complete assessment of system stability during network operation.

An alternative method using a direct stability analysis approach is presented in the current chapter for the fast assessment of system stability. The method is based on Lyapunov's second method for fast classification of network contingencies and identification of critical system settings. The presented method achieves fast stability assessment by eliminating explicit numerical integration in the post-fault period while deriving the system state of stability. Specifically, the potential energy boundary surface (PEBS) approach is applied for the characterization of the region of stability. The analysis approach combines the maximum potential energy criterion and the computation of the directional derivative of the potential energy function in order to reliably estimate the region of stability without computing controlling unstable equilibrium points. The method is tested using standard test networks and network representations of real transmission grids for evaluation of accuracy and performance. Results from the time-domain simulation approach are used as a benchmark in the validation process.

6.1 Formulation of the Direct Stability Assessment Method

The direct stability analysis approach presented in the current chapter is based on Lyapunov's second method. The main requirements in this stability analysis approach are; a transient energy function derived for a specific network structure, and a defined region of stability. A transient energy function is derived from the equations representing the power system, while the region of stability is estimated using a critical energy approximation defined for a given disturbance and a corresponding post-fault equilibrium point. The present section describes the assumptions considered in deriving the power system mathematical representation, the formulation of the transient energy function, and the approach developed for estimation of the region of stability.

The current chapter is based on the work published in [164].

M. Kyesswa, H. K. Çakmak, L. Gröll, U. Kühnapfel, and V. Hagenmeyer, "A Hybrid Analysis Approach for Transient Stability Assessment in Power Systems," in *2019 IEEE Milan PowerTech*, Milan, Italy, pp. 1–6, June 2019.

6.1.1 System State Model

The following simplifying assumptions are considered in deriving the power system mathematical representation used in the method presented in the current chapter: 1) Generators are represented by a classical model with a constant voltage behind a transient reactance; 2) Mechanical power input to the generator is constant; 3) Loads are represented as constant impedances. With these simplifications, the dynamics of the generator are represented by the equation of motion in Equations (2.3) and (2.4) expressed in the synchronous reference frame.

For the formulation of the transient stability solution in the presented method, the differential equations in (2.3) and (2.4) are redefined in the center of inertia (COI) reference frame, where all the angles and speeds are measured with respect to the center of inertia. This formulation eliminates large mismatches in power between the reference machine and the rest of the system during changing operating conditions. The COI is defined by the center of angle δ_0 and center of speed ω_0 given as

$$\delta_0 = \frac{1}{M_T}\sum_{i=1}^{n} M_i\delta_i;\ \ \omega_0 = \frac{1}{M_T}\sum_{i=1}^{n} M_i\omega_i;\ \ \text{where}\ \ M_T = \sum_{i=1}^{n} M_i.$$

The resulting dynamics of the i^{th} generator expressed in the center of inertia reference frame are given by [104, 102]

$$\dot{\theta}_i = \tilde{\omega}_i, i = 1, 2, ..., n \tag{6.1}$$

$$M_i\dot{\tilde{\omega}}_i = P_i - \sum_{j=1,j\neq i}^{n}\Big[C_{ij}\sin\theta_{ij} + D_{ij}\cos\theta_{ij}\Big] - \frac{M_i}{M_T}P_{COI} = f_i(\theta) \tag{6.2}$$

where $f_i(\theta)$ is a function representing the accelerating power in the COI reference. The state variables $\theta_i = \delta_i - \delta_0$ and $\tilde{\omega}_i = \omega_i - \omega_0$ are the relative angle and the relative rotor speed transformed into the COI reference, respectively; δ_i and ω_i are the internal rotor angle and speed deviation of the i^{th} generator in synchronous reference frame, respectively. The relative rotor angle between machines i and j is $\theta_{ij} = \theta_i - \theta_j$ in a system of n machines. M_i is the moment of inertia of the i^{th} machine and M_T is the total inertia of the system. The terms C_{ij} and D_{ij} are derived from the expression of the electrical power output given by

$$P_{ei} = E_i^2 G_{ii} + \sum_{j=1,j\neq i}^{n}\Big[E_iE_jB_{ij}\sin\theta_{ij} + E_iE_jG_ij\cos\theta_{ij}\Big]$$

whereby $C_{ij} = E_iE_jB_{ij}$; $D_{ij} = E_iE_jG_{ij}$; E_i and E_j are the voltages behind direct axis transient reactance of the respective machines at nodes i and j; The terms B_{ij} and G_{ij} are the transfer susceptance and conductance of the branch between machines i and j in the reduced admittance matrix of the network, respectively. The variables P_i and P_{COI} in Equation (6.2) are defined as

$$P_i = P_{mi} - E_i^2 G_{ii};\;\; P_{COI} = \sum_{i=1}^{n} P_i - 2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} D_{ij}\cos\theta_{ij}$$

where P_{mi} is the mechanical power input and P_{COI} is the power of the center of inertia. The state equations (6.1) and (6.2) are used to describe the system dynamics in the following three states: Pre-fault state – system before the occurrence of the disturbance; Faulted state – system during the disturbance; Post-fault state – system after clearance of the disturbance. The equations in each of the states only differ in the transfer elements of the branches as a result of the difference in the internal structure of the network in the three operating states.

6.1.2 Transient Energy Function

Applying a Lyapunov-based method for power system stability assessment requires definition of a transient energy function for the network under analysis. For the method presented in the current chapter, the energy function is based on the general expression of the transient energy function derived by Athay et al. [104] given as

$$V(\theta,\tilde{\omega}) = \frac{1}{2}\sum_{i=1}^{n} M_i\tilde{\omega}_i^2 - \sum_{i=1}^{n} P_i(\theta_i - \theta_i^s) - \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[C_{ij}\Big(\cos\theta_{ij} - \cos\theta_{ij}^s\Big) - \int_{\theta_i^s+\theta_j^s}^{\theta_i+\theta_j} D_{ij}\cos\theta_{ij}\,d(\theta_i+\theta_j)\right] \quad (6.3)$$

In Equation (6.3), the terms C_{ij} and D_{ij} refer to the post-fault system parameters, since the transient energy function is constructed for the post-fault system. The variable $\theta_{i,j}^s$ is the post-fault stable equilibrium point (SEP) for machines i, j. At the equilibrium point, the rotor speed deviation $\tilde{\omega}_i^s = 0$ since all rotor speeds are equal to a reference value. The system equilibrium point is therefore defined as $(\theta^s, 0)$. The point $(\theta_i, \tilde{\omega}_i)$ defines the actual system state of the i^{th} machine at a given time.

The transient energy function in Equation (6.3) represents a summation of the system kinetic energy V_{KE} and potential energy V_{PE} and can be expressed as $V(\theta,\tilde{\omega}) = V_{KE}(\tilde{\omega}) + V_{PE}(\theta)$. The kinetic energy is due to the rotation of the rotor and is represented by the first term on the right hand side. The total potential energy is due to a combination of the remaining three terms on the right hand side representing the potential energy as a result of; the rotor position, change in stored magnetic energy in the branch between generators i and j, and change in dissipated energy in the branch between generators i and j, respectively. This breakdown of the total energy is important for the derivation of the region of stability boundary applied in the presented method as described in the next subsection.

The value of the potential energy due to the fourth term in Equation (6.3) depends on the branch transfer conductance. Therefore, this term is path dependent since it results from the non-conservative dissipation of energy in the branch. In order to compute the actual value of

the fourth term, the system trajectory is required defining the path from the stable point $\theta^s_{i,j}$ to the final fault clearing point $\theta_{i,j}$. However, the exact trajectory is unknown in the presented method. An approximate solution is therefore applied for the path dependent integral in the transient energy function, using the linear trajectory approximation of the system trajectory in the angle space. In this case, a linear trajectory is assumed to represent the system from the stable equilibrium point $\theta^s_{i,j}$ to the angle $\theta_{i,j}$ at the stability boundary [104]. The additional integrand arising due to the above approximation is solved using the trapezoidal method. Equation (6.4) is the resulting expression for the approximation of the change in dissipated energy in the system [104, 165].

$$\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\int_{\theta^s_i+\theta^s_j}^{\theta_i+\theta_j} D_{ij}\cos\theta_{ij}\,d(\theta_i+\theta_j) = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} D_{ij}\frac{\theta_i+\theta_j-\theta^s_i-\theta^s_j}{\theta_{ij}-\theta^s_{ij}}(\sin\theta_{ij}-\sin\theta^s_{ij}) \quad (6.4)$$

6.1.3 Derivation of Stability Region

The objective of the direct method is to estimate the system's state of stability by locating the initial state of the post-fault system with respect to a defined region of stability. Having derived the transient energy function in Section 6.1.2, the next vital step is to define the region of stability. In the presented method, an approximate measure using a critical energy value is considered to estimate the region of stability, as assumed in power system applications [166]. The supposition in this estimation is that a system attains a critical energy at the boundary of the stability region. Thereby, the interior of the stability region is defined by the inequality $V(\theta,\tilde{\omega}) < V_{cr}$; where $V(\theta,\tilde{\omega})$ is the system transient energy at point $(\theta,\tilde{\omega})$ and V_{cr} is the critical energy [166]. This approximation reduces the problem of determining the region of stability to estimation of critical energy.

Estimation of the critical energy in the presented method is based on the potential energy boundary surface (PEBS) method [104, 103]. The PEBS characterization of the stability region is founded on the notion that the transient energy function is composed of a kinetic energy component and potential energy component as shown in Equation (6.3). In terms of system stability, the total kinetic energy gained due to acceleration of the machines in the fault-on period should be transformed into potential energy in the post-fault period in order to maintain synchronous operation in the entire system. Based on this concept, the PEBS approach simplifies the problem of determining critical energy to estimating the maximum potential energy that the system can absorb in the post-fault state.

A new methodology is derived for determining the critical energy in the presented method. The methodology combines the maximum potential energy criterion and validation of the true stability boundary using the directional derivative of the potential energy function. The combined approach is used to determine the global maximum potential energy of the full system. The derived methodology is described in the following subsections.

Estimation of Maximum Potential Energy

For the estimation of the maximum potential energy, the potential energy function $V_{PE}(\theta)$ is derived from the transient energy function in Equation (6.3). Computation of the potential energy function begins from the point of fault occurrence. The function is followed at every time step during the fault-on period until the maximum potential energy $V_{PE,max}$ is reached. This point defines the boundary of stability and gives a good estimate of the critical energy, as described in the PEBS method proposed in [103]. However, it should be noted that the first maximum value of the potential energy does not always indicate the true maximum point in the full state space. This is illustrated in Figure 6.1. Therefore, an additional criterion is applied to prove the existence of the true maximum critical point of the system within a specified time.

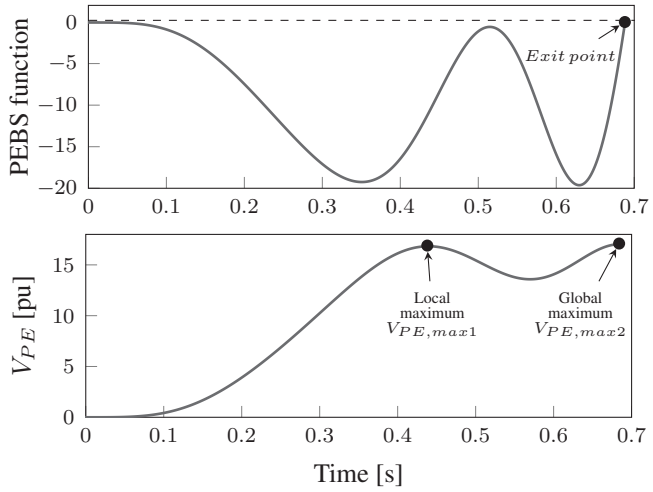


Figure 6.1: Combined approach for estimation of critical energy; Monitoring the potential energy (V_{PE}) and directional derivative of the potential energy function (PEBS function).

Test for Global Maximum Potential Energy Point

The presented method applies the directional derivative of the potential energy function along the fault-on trajectory in the angle space to ascertain the true crossing point of the stability boundary. The estimation is mathematically represented by the dot product $f^{\mathsf{T}}(\theta) \cdot (\theta - \theta^s)$ [165], where $f(\theta)$ is the acceleration power in Equation (6.2) of the fault-on system. This product is referred to as the PEBS function in the rest of the current thesis. Thereby, the region of stability is defined as follows [165]:

- $f^{\mathsf{T}}(\theta) \cdot (\theta - \theta^s) < 0$ within the stability region,
- $f^{\mathsf{T}}(\theta) \cdot (\theta - \theta^s) = 0$ at the PEBS crossing point, and
- $f^{\mathsf{T}}(\theta) \cdot (\theta - \theta^s) > 0$ outside the stability region.

In the presented combined approach, the directional derivative and the potential energy function are monitored at every time step until a change in sign is observed in the PEBS function, i.e. from negative to positive; this point defines the PEBS boundary crossing. Once the exit point is determined, the maximum of the potential energy profile from the stable equilibrium point to the crossing point gives an estimate of the critical energy $V_{cr} = V_{PE,max}(\theta)$.

Figure 6.1 illustrates the estimation of the critical energy using the presented method. As depicted in the figure, the first maximum potential energy $V_{PE,max1}$ represents a local maximum point of the potential energy. The additional test using the directional derivative identifies another maximum point $V_{PE,max2}$ before the exit of the stability region. In this case, the maximum of the two detected points defines the global maximum potential energy, which gives the estimate of the critical energy; and thus defines the stability boundary. This method eliminates the computationally challenging requirement of computing the controlling unstable equilibrium points during the assessment process as applied in the methodology presented in [167].

6.2 Proposed Assessment Process

The current section describes the computational algorithm in the presented method for the fast stability assessment and classification of network contingencies. The method is implemented in the Matlab environment. At the input stage of the computation method, the structure of the network to be analyzed is defined in the form of Matpower case files, similar to the structures used in the time-domain simulations described in Chapter 3. In addition to the network structure, dynamic parameters of the machines in the system and simulation settings – representing the protection settings for clearing the disturbances – are also defined at the input stage. The following subsections describe the main steps of the analysis method.

6.2.1 Initialization

The presented method starts by determining the initial operating conditions of the network. This initialization procedure is performed using the pre-fault network parameters to determine the steady state values of the system. Like in the time-domain simulations, the main computation at the initialization stage is the calculation of the power flow equations. In the developed method, the Matpower power flow algorithm [33] is applied for the required power flow calculations. All the time-derivatives of the dynamic state variables are set to zero at the initialization stage. With this, the power flow results are used to solve the resulting system equations in order to find the initial values of the remaining algebraic variables and the initial dynamic state variables.

For the simplified system state model applied in the presented direct computational method as described in Section 6.1.1, the power flow results are used to obtain the following variables: State variable – the initial internal generator angle δ_0 and steady state rotational speed ω_0 for all the connected machines in the system; Algebraic variables – initial generator internal voltage E_0, initial machine armature current injections I_{a0}, and the mechanical input power P_{m0}. In addition, the values of the constant impedance loads are computed using initial active and reactive power, and the voltages at the load buses.

6.2.2 Formulation of Reduced Network

The parameters of the system state model described in Equations (6.1) and (6.2) are defined in reference to only generator nodes. This form of system representation is referred to as the reduced network since it eliminates all non-generator nodes from the network model. The process of forming the reduced network is described in the present section.

In the first stage of the network reduction process, system parameters are set up for the full network for the three sections of the simulation: pre-fault, fault-on, and post-fault sections. This involves modifying the network to include the constant impedance loads and generator internal reactances. The network is modified as follows in the presented method: internal generator nodes are created and connected to the terminal nodes via the generator direct axis transient reactance $x_d^{'}$; the constant load impedances are connected between the load nodes and the network reference node. Using the modified network, an admittance matrix Y is formed and the constant impedance loads directly added to the diagonals of the matrix at the corresponding nodes. The admittance matrix is formed using the *makeYbus* function from the Matpower library [33].

For the formulation of the fault-on system parameters, the network parameters are adjusted according to the simulated fault and its location. The admittance matrix is modified using an event handling process, where the changes in network structure are considered following the fault and the admittance matrix is reformulated. The same procedure applies for the post-fault system parameters, following a change in network structure beyond the fault clearing time. Possible changes in the network in the post-fault state include line switching to clear a fault, and changes in load or generation.

In the second stage, a reduced network matrix Y_{red} is formed from the constructed full system admittance matrix Y by eliminating all nodes (r) except the internal generator nodes (n) using Kron reduction [168]. The formulation of the reduced network matrix for each of the simulation states is given by

$$Y_{red} = Y_{nn} - Y_{nr} Y_{rr}^{-1} Y_{rn} \tag{6.5}$$

where Y_{red} is an $n \times n$ Kron-reduced network matrix, Y_{nn} is a diagonal matrix of the sum of admittances connected to the n generator nodes, $Y_{nr} = Y_{rn}$ are elements of the admittance matrix between the n^{th} generator node and the r^{th} non-generator node, Y_{rr} consists of the sum of admittances connected to the r non-generator nodes. The parameters G_{ii}, C_{ij} and D_{ij} in the system state equation (6.2) and in the transient energy function in Equation (6.3) are derived from the Kron-reduced matrix.

The network reduction is carried out at every occurrence of network modification. Analysis of Equation (6.5) shows that the network reduction process scales with more than quadratic complexity, dominated by the inversion of matrix Y_{rr}. In the proposed method, the formulation of the reduced network admittance matrix is simplified using the compensation method based on the matrix inverse lemma described in [169]. The compensation method simplifies the inverse computation using a relation between the inverse $Y_{rr,0}^{-1}$ from the original network admittance matrix and the inverse $Y_{rr,mod}^{-1}$ from the modified admittance matrix given by

$$Y_{rr,mod}^{-1} = Y_{rr,0}^{-1} - \left[Y_{rr,0}^{-1} \cdot M \cdot \left(\Delta y^{-1} + M^{\mathsf{T}} \cdot Y_{rr,0}^{-1} \cdot M \right)^{-1} \cdot M^{\mathsf{T}} \cdot Y_{rr,0}^{-1} \right] \tag{6.6}$$

where Δy is a matrix defining the modifications in the parameters of the original admittance matrix, and M is a connection matrix, which is built based on the type of modification. In the present thesis, the branch oriented modification is considered, where a column vector is defined for each modified branch in matrix M with $+1$ and -1 entries on the respective nodes defining the branch connection and 0 elsewhere. For a short circuit connection of a bus to ground, which is represented by connection of a shunt on the corresponding bus, an entry $+1$ is defined for the relevant bus position in the M matrix. Further details of the formulation of the branch-oriented modification and of the alternative node-oriented modification are given in [170]. Formulation of the modified network reduced admittance matrix $Y_{red,mod}$ is thus given by

$$Y_{red,mod} = Y_{nn} - Y_{nr} Y_{rr,mod}^{-1} Y_{rn}. \tag{6.7}$$

6.2.3 Initial Elimination: Unstable Equilibrium Points

As shown in Equation (6.3), the application of the computation of the transient energy function depends on the existence of a stable equilibrium point of the resulting post-fault network. Therefore, the presented direct analysis method begins by determining the post-fault equilibrium point. Given that the time derivatives in Equations (6.1) and (6.2) are zero at the equilibrium point, the post-fault equilibrium point is determined by solving the accelerating power equation $f_i(\theta) = 0$, for $i = 1, 2, \ldots, n$. In this method, the solution of the system equation is obtained using Newton's method.

Furthermore, it is important to determine the state of the post-fault equilibrium point, i.e. whether the equilibrium point of the system is stable or unstable after the disturbance is cleared. In the presented method, the state of stability is tested using the linearization principle about the calculated post-fault equilibrium point, and applying the eigenvalue analysis technique on the system dynamics' state Jacobian matrix as described in the following.

Linearizing the system state equations in (6.1) and (6.2) about a general equilibrium point $(\theta_s, 0)$ results in the differential equations in variables $\Delta\theta_i$ and $\Delta\tilde{\omega}_i$ for each machine given by Equation (6.8). The differential equations in (6.8) are written in state matrix form given by Equation (6.9), where I represents an identity matrix. For an n machine system, the state matrix is of the order $2n \times 2n$.

$$\begin{aligned} \Delta\dot{\theta}_i &= \Delta\tilde{\omega}_i, \; i = 1, 2, \ldots, n \\ M_i \Delta\dot{\tilde{\omega}}_i &= \frac{\partial f_i}{\partial \tilde{\omega}_i} \Delta\tilde{\omega}_i + \frac{\partial f_i}{\partial \theta_i} \Delta\theta_i \end{aligned} \tag{6.8}$$

$$\begin{bmatrix} \Delta\dot{\theta}_i \\ \Delta\dot{\tilde{\omega}}_i \end{bmatrix} = \begin{bmatrix} 0 & I \\ M_i^{-1} \frac{\partial f_i}{\partial \theta_i} & M_i^{-1} \frac{\partial f_i}{\partial \tilde{\omega}_i} \end{bmatrix} \begin{bmatrix} \Delta\theta_i \\ \Delta\tilde{\omega}_i \end{bmatrix} \tag{6.9}$$

By analyzing the eigenvalues of the state matrix in Equation (6.9), the state of the equilibrium point is characterized as follows: The equilibrium point is stable if all the eigenvalues λ_x of the state matrix have negative real parts, i.e. $\Re\{\lambda_x\} < 0$, for all x. Otherwise, the equilibrium point is unstable if any of the eigenvalues has a positive real part, i.e. $\Re\{\lambda_x\} > 0$, for any x [17].

The above procedure is used as a first elimination step in the presented method. The method categorizes the contingencies at this stage as follows: If the post-fault equilibrium point does not exist from the solution of the accelerating power equation or if the post-fault equilibrium point is unstable, the corresponding switching operation is considered to be an infeasible case, where the stability of the system cannot be analyzed using the direct method. Such cases are referred to as unclassifiable contingencies in the rest of the present thesis. Otherwise, a contingency with a stable post-fault equilibrium point can be further analyzed as described in the next section.

6.2.4 Classification of Contingencies

The main purpose of the presented direct method is the fast classification of network contingencies. This is performed for the cases with stable post fault equilibrium points identified from the previous step. The classification of the contingencies at this stage is in two categories: (i) Noncritical contingency, in which the system is identified to remain stable following the disturbance; (ii) Critical contingency, where the disturbance drives the system into an unstable operating condition. The classification procedure is described in the following.

At the classification stage, the presented direct method begins by estimating the system fault-on trajectory. The fault-on trajectory is estimated by numerical integration using the system equations in the fault-on state. In the developed method, similar numerical integration algorithms as in the time-domain simulation approach are applied at this stage. Specifically, the explicit fourth order Runge-kutta integration method is applied to derive the fault trajectory. Each time step of the integration gives a value of internal generator rotor angle θ_t and rotational speed $\tilde{\omega}_t$, resulting in the system operating point $(\theta_t, \tilde{\omega}_t)$. This process is continued until the fault clearing time.

The system trajectory derived in the first step is necessary for defining the location of the system with respect to the region of stability. However, the region of stability in the present method is defined based on the system potential energy function. Therefore, the system state $(\theta_t, \tilde{\omega}_t)$ at each time step of the numerical integration is used to compute the value of the potential energy $V_{PE}(\theta_t)$. At the same time, the corresponding value of the dot product $f^{\mathsf{T}}(\theta_t) \cdot (\theta_t - \theta^s)$ is computed in order to monitor the directional derivative of the potential energy function. The value of θ_t used in the computation of these variables is obtained from the fault-on angle, whereas the admittance matrix is based on the post-fault system.

As described in Section 6.1.3, the PEBS crossing is reached at the zero crossing of the directional derivative, i.e. at a point where $f^{\mathsf{T}}(\theta_t) \cdot (\theta_t - \theta^s) = 0$. After determining this point, reference is made to the potential energy to determine the critical energy as the maximum value of the potential energy profile. This defines an estimate of the region of stability for the post-fault system.

The present method requires that the state at each time step during the integration of the fault-on system is tested to determine whether it lies within the stability region. This is defined by the inequality $V(\theta_t, \tilde{\omega}_t) < V_{cr}$ as described in Section 6.1.3. Thereby, this requires the value of the transient energy $V(\theta_t, \tilde{\omega}_t)$ at successive times using the fault-on θ_t and $\tilde{\omega}_t$ obtained from the step by step integration of the fault-on trajectory. The classification of the contingencies is then defined as follows: If the transient energy is less than the critical energy, i.e. $V(\theta_t, \tilde{\omega}_t) < V_{cr}$, at

the initial state of the post-fault system trajectory, the simulated contingency is considered to be a stable (Noncritical contingency) for the defined fault clearing time. Otherwise, if the contingency results in a gain in transient energy greater than the critical energy, i.e. $V(\theta_t, \tilde{\omega}_t) > V_{cr}$, at the initial state of the post-fault trajectory or before the fault clearing time ($t < t_{cl}$), the contingency is considered to be unstable (Critical contingency).

In addition to the fast identification of system stability, important parameters are derived from the analysis using the presented method. These include fault critical clearing time (CCT) and stability margins. The critical clearing time of a disturbance is estimated at the point where the transient energy is equal to the critical energy, i.e. $t_{cct} = t(V(\theta_t, \tilde{\omega}_t) = V_{cr})$. The stability margin of a critical contingency defines the distance – in terms of energy – from stability at the time of fault clearance. The stability margin is given by $S_m = V_{cr} - V_{cl}$, where V_{cl} is the transient energy of the system at the clearing time. Such information is necessary for comparing structural configurations.

Figure 6.2 illustrates the stability analysis process in the presented direct method. The time t_{end} in the simulation process is set according to the required fault clearing time (t_{cl}). In practical applications, this time corresponds to the minimum time required for the protection systems to isolate the fault from the system. It is important to note that noncritical contingencies take up more computational time than critical contingencies since they are tested for the whole duration $t_{end} = t_{cl}$ during the selection process to confirm system stability. The simulation of critical contingencies is terminated when the critical energy is reached.

Furthermore, it is worth noting that the present direct method uses an internal integrator for the derivation of the system trajectory. With this, the system variables are available during the fault-on simulation and the trajectories of the system variables are directly applied to obtain transient energy, potential energy and directional derivative during the screening process. This is critical for the speed of analysis.

6.3 Evaluation of Accuracy and Performance

The current section presents test simulations in order to verify the accuracy and performance of the presented direct analysis approach. The time-domain simulation approach described in Chapter 3 is used to provide the benchmark results for evaluating the accuracy of the direct method. Comparison of the two approaches is carried out in terms of the ability to correctly categorize network contingencies and accuracy of estimating the critical clearing time of faults.

In the first test case, the simulated network structure is the IEEE 9-bus test network. This simple case is mainly considered to verify the accuracy of the critical clearing time estimates against the time-domain simulation approach based on numerical integration. The second test system is the IEEE 118-bus test network. The main aim in this simulation case is to illustrate the stability analysis procedure in the direct method, including the eigenvalue analysis to test the state of the equilibrium point, the monitoring process to determine the crossing point of the stability boundary and estimation of the critical clearing time. The estimated critical clearing times by the direct method for selected contingencies are verified against the results from the time-domain simulation approach. Furthermore, the reliability and performance of the presented method are evaluated and compared to the state-of-the-art SIME-based method presented

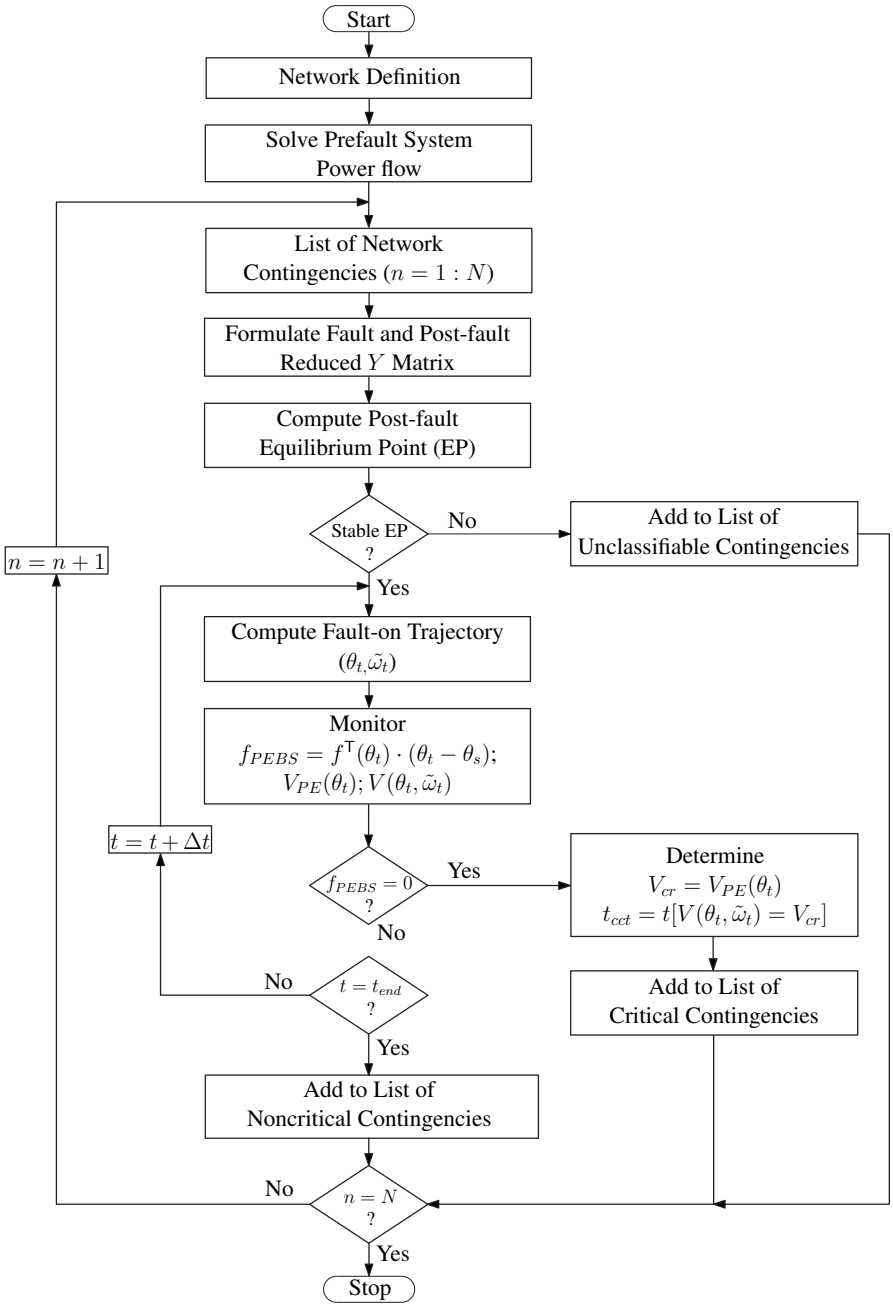


Figure 6.2: Direct method stability analysis process

in [112]. The networks applied for evaluation of reliability and performance are summarized in Table 6.1. Case68, Case118 and Case1354 are standard IEEE test networks, but modified by adding internal buses for generators. CaseBW, and CaseDE represent the modified networks of the Baden-Württemberg state and the Germany transmission grids, respectively.

The tested systems presented in the current section are limited to symmetrical and balanced network configurations. A contingency is defined by the fault type, fault location and action

Table 6.1: Test networks with varying complexity

Test Network	Gens	Buses	Branches	No. of Contingencies
Case68	16	68	83	186
CaseBW	17	150	226	551
Case118	54	172	240	490
Case1354	260	1614	2251	5336
CaseDE	511	2146	3179	6971

taken to clear the fault. The mode of clearance of the fault defines the post-fault network structure. In the presented test cases, the analyzed contingency is a three phase-to-ground short circuit fault located on or near a bus, followed by the required switching operation to clear the fault. The following operations are considered to clear the fault: Switching out a line connecting to the fault bus, which results in a change in network structure in the post-fault state; or a self-clearing fault, where no switching action is taken and the resulting post-fault network structures is similar to the pre-fault network.

6.3.1 Test case I: 9-Bus System

The network structure of the IEEE 9-bus test system used in the first test case is described in Section 3.3.1. For the simulations in the current section, the generators are represented by the classical model with constant input mechanical power and constant field excitation. The main parameters of the generators are defined from the 9-bus structure described in [18]. The network consists of nine buses and nine branches. A branch in the network is defined by the buses on either end expressed in the form "from–to" [33], where "from" is the bus index on the sending end and "to" is the bus index on the receiving end of the branch.

The 9-bus network is set up with two sets of generator parameters. This is performed in order to analyze the variation of system stability with changing system parameters and test the reliability of the presented method in presence of changing network conditions. For illustration purposes, the fault clearing time in each case is set to $t_{cl} = 2\,\mathrm{s}$ to represent the maximum time the fault can be present on the network. The step-by-step numerical integration for deriving the fault-on trajectory is computed with a fixed step size of $1\,\mathrm{ms}$. The set up and results with the two sets of parameters are described in the following.

Parameter Set 1:

The first set of generator parameters is given in Table 6.2. The listed parameters are defined as follows: H is the inertia constant, D is the damping constant, x_d is the steady state reactance, and $x_d^{'}$ is the transient reactance. The generator column gives the bus indices to which the generators are connected. All parameters are expressed in per unit (pu) values.

Table 6.3 shows the simulation results comparing the critical clearing time (CCT) obtained using the presented method (CCT_{DM}) and the time-domain simulation approach (CCT_{TDS}). The table includes the list of simulated contingencies – in terms of the fault location (fault bus)

Table 6.2: Set 1 generator parameters

Generator	H	D	x_d	$x_d^{'}$
1	23.64	0.02	0.146	0.0608
2	6.4	0.02	0.8958	0.1198
3	3.01	0.02	1.3125	0.1813

Table 6.3: List of contingencies and critical clearing times for set 1 generator parameters

Contingency	Fault Bus	Line Trip	CCT_{DM}(s)	CCT_{TDS} (s)
1	4	–	0.326	0.368
2	4	5–4	0.316	0.365
3	4	6–4	0.313	0.354
4	5	–	0.411	0.478
5	5	5–4	0.413	0.488
6	5	7–5	0.319	0.401
7	6	–	0.461	0.572
8	6	6–4	0.453	0.560
9	6	6–9	0.393	0.499
10	7	–	0.233	0.254
11	7	5–7	0.179	0.196
12	7	8–7	0.199	0.200
13	8	–	0.327	0.371
14	8	7–8	0.313	0.308
15	8	9–8	0.324	0.337
16	9	–	0.259	0.297
17	9	9–6	0.242	0.258
18	9	9–8	0.240	0.271

and fault clearing type (line trip) – and the critical clearing time. The line trip column lists the respectively switched lines to clear the fault, represented in terms of "from–to" bus indices. The dash "–" entry in the line trip column indicates cases with no line switching action to clear the fault, i.e. the fault is considered to be self-clearing in such cases.

The test case consists of 18 contingencies as shown in Table 6.3. The contingencies are classified as critical or noncritical using the relationship between the estimated critical clearing time (t_{cct}) and the set simulation time ($t_{cl} = 2.0s$). The time t_{cl} represents the maximum duration of the fault before the operation of protection devices. The classification of the contingencies is therefore summarized as follows: Critical contingency (unstable), if $t_{cct} < 2.0$ s; Non-critical contingency (stable), if $t_{cct} > 2.0$ s.

Analyzing the results in Table 6.3, it is observed that the critical clearing times using the presented method CCT_{DM} are less than 2 s for all the contingencies. The presented method therefore categorizes all contingencies as critical. The time-domain simulation approach also derives a similar classification of the contingencies as shown in the CCT_{TDS} column. Both

approaches deduce that all the tested contingencies result in unstable network operation within the fault simulation time of 2 s. In terms of accuracy, the critical clearing times estimated by the presented direct method are in close agreement with the estimates by numerical integration in time domain simulations. However, the presented direct method is observed to provide under estimations compared to the time-domain simulations. This difference can be attributed to the conservative nature of the PEBS-based direct method.

Parameter Set 2

Modified generator parameters are applied to the IEEE 9-bus in this simulation case in order to further test the reliability of the presented direct method. The new generator parameters are given in Table 6.4. For testing purposes, the parameters are similar for all generators in this test scenario.

Table 6.4: Set 2 of generator parameters

Generator	H	D	x_d	$x_d^{'}$
1	5.74	0.02	0.14	0.14
2	5.74	0.02	0.14	0.14
3	5.74	0.02	0.14	0.14

Table 6.5: List of contingencies and critical clearing times for set 2 generator parameters

Contingency	Fault Bus	Line Trip	CCT_{DM}(s)	CCT_{TDS} (s)
1	4	–	1.397	1.698
2	4	5–4	1.424	1.724
3	4	6–4	1.361	1.679
4	5	–	> 2	> 2
5	5	5–4	> 2	> 2
6	5	7–5	> 2	> 2
7	6	–	> 2	> 2
8	6	6–4	> 2	> 2
9	6	6–9	> 2	> 2
10	7	–	0.259	0.276
11	7	5–7	0.23	0.256
12	7	8–7	0.212	0.233
13	8	–	0.41	0.45
14	8	7–8	0.337	0.393
15	8	9–8	0.38	0.428
16	9	–	0.725	0.921
17	9	9–6	0.674	0.863
18	9	9–8	0.722	0.899

Table 6.5 summarizes the results of the calculated critical clearing times using the second parameter set. The first observation from the results shown in Table 6.5 is that the new system parameters result in a change in classification of contingencies in the presented method and time-domain simulations. In this case, both approaches show that the network remains in stable operation for six of the simulated contingencies (i.e. contingencies 4 to 9). This implies that the critical clearing time for such contingencies is beyond the set duration of the fault, i.e. $t_{cct} > 2.0\,\mathrm{s}$. These contingencies are thus categorized as noncritical. For the remaining contingencies, a corresponding critical clearing time is estimated in both approaches, and are therefore categorized as critical contingencies. The critical clearing times estimated from the direct method further show a close agreement with the time-domain simulation results. Therefore, the presented direct method is able to accurately and reliably select and classify network contingencies with close accuracy to the numerical integration-based time-domain simulations.

6.3.2 Test case II: 118-Bus System

In the present section, the presented method is further tested in a second network to evaluate the performance of the algorithm with increasing network complexity. The test network applied in this case is the IEEE 118-bus system. The network structure and parameters are based on the model described in Matpower [33]. The network is modified by adding internal nodes at the generator terminals, which results in a modified system with 172 nodes, and 240 branches.

The total number of contingencies in the IEEE 118-bus test network is 490 as shown in Table 6.1. For testing purposes, the fault duration is set to $2.0\,\mathrm{s}$ for each contingency. In this test case, the classification of the contingencies by the presented direct method is summarized as follows: Unclassifiable contingencies – 2; Noncritical contingencies – 122; Critical contingencies – 366. The aim in the current section is to illustrate the stability assessment process in the direct method. One contingency is considered from each category for purposes of illustration as described in the following.

Unclassifiable Contingency: Fault Bus 9, Line Trip 9–10

For an unclassifiable case, the following contingency is selected: A fault applied on bus 9 and cleared by switching out the line between bus 9 and bus 10. As a first step, the post-fault equilibrium point is computed and tested for stability using the eigenvalue analysis approach. Figure 6.3 shows the plot of the eigenvalues in the complex plane. The results in Figure 6.3 show that the state matrix at the post-fault equilibrium point has some eigenvalues with positive real parts. Therefore, the post-fault equilibrium point in this case is identified to be unstable. This implies that the contingency is unclassifiable, whereby system stability cannot be further analyzed using the direct approach. For the IEEE 118-bus network analyzed in this case, two contingencies are identified as unclassifiable by the direct method.

Noncritical Contingency: Fault Bus 97, Line Trip 96–97

The following contingency is considered to illustrate the analysis of a noncritical case: A fault applied on bus 97, cleared by switching out the line between bus 96 and bus 97. The eigenvalues

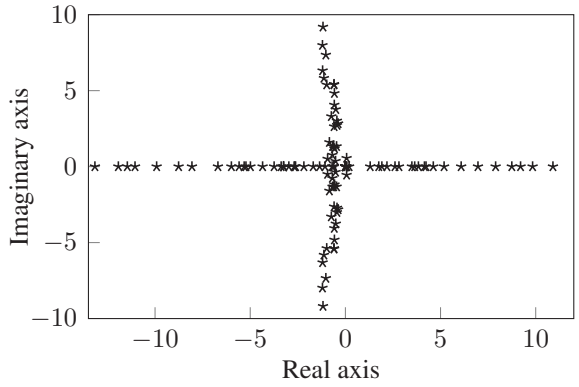


Figure 6.3: Eigenvalues for post-fault system without line 9–10. State matrix at post fault equilibrium point has eigenvalues with positive real parts; thus unstable equilibrium point.

of the resulting state matrix at the post-fault equilibrium point are plotted in the complex plane as shown in Figure 6.4. According to the plot in Figure 6.4, the state matrix has no eigenvalues with positive real parts in this case. This implies that the post-fault equilibrium point is stable and the direct method can proceed with the actual stability analysis of the contingency.

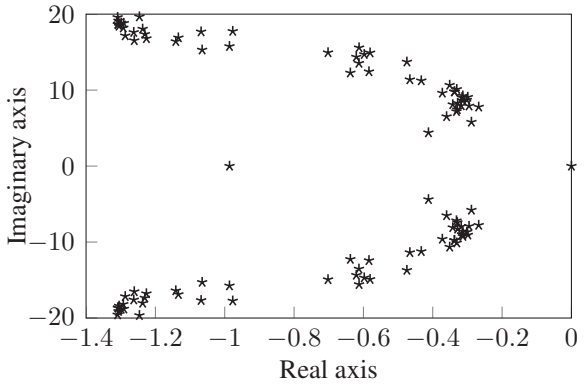


Figure 6.4: Eigenvalues for post-fault system without line 96–97; State matrix at post fault equilibrium point has no eigenvalues with positive real parts; thus stable equilibrium point.

The next step of the analysis in the direct method is to locate the stability boundary crossing by monitoring the directional derivative, potential energy and transient energy of the system. Figure 6.5 shows the results of the monitoring process. It can be observed from Figure 6.5 that the PEBS function representing the directional derivative is always less than zero for the whole simulation time of 2.0 s. From the definition of the stability region in Section 6.1.3, this shows that the system trajectory remains within the stability region. This contingency is therefore classified to be stable and noncritical for the defined simulation duration.

As part of the validation process, the same contingency is simulated using the time domain simulation approach to analyze the system response for a fault clearing time of 2.0 s. The response of the generator rotational speed is shown in Figure 6.6. The step-by-step simulation results in Figure 6.6 show that there is a momentary acceleration of the generators following

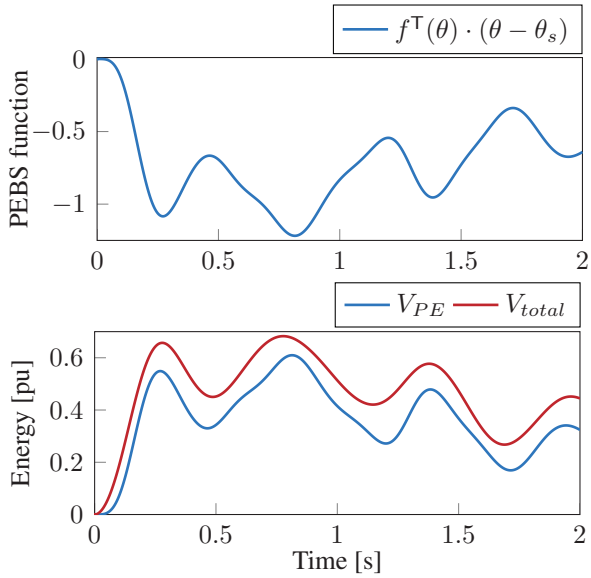


Figure 6.5: Monitoring PEBS function, total energy (V_{total}) and potential energy (V_{PE}) for a noncritical case

the fault. However, the generators accelerate almost in synchronism during the fault and return to a common operating point in the post-fault period. From this observation, the time domain simulation approach classifies the contingency as stable and noncritical for a fault duration of $2.0\,\mathrm{s}$, similar to the presented direct method.

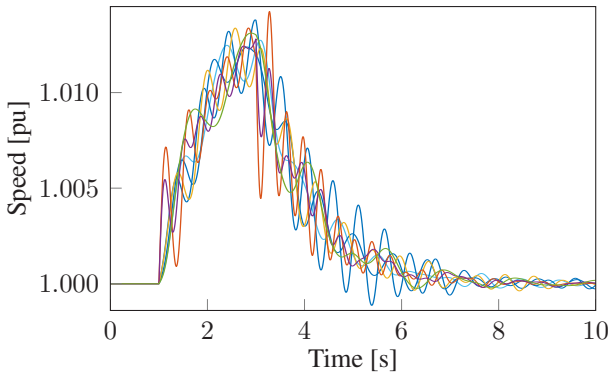


Figure 6.6: Generator rotational speed response for fault clearing time $t_{cl} = 2.0\,\mathrm{s}$; generators always remain in synchronism after the fault.

Critical Contingency: Fault bus–46, Line Trip 46–47

The critical contingency considered for further illustration of the assessment process in the direct method is defined as follows: A fault on bus 46, cleared by switching out the line between bus 46 and bus 47. The initial analysis using the eigenvalue approach shows that the post-fault equilibrium point is stable. The plot of the eigenvalues of the state matrix is similar to that

shown in Figure 6.4. As a next step, the directional derivative, potential energy and transient energy function are monitored during the fault period as shown in Figure 6.7.

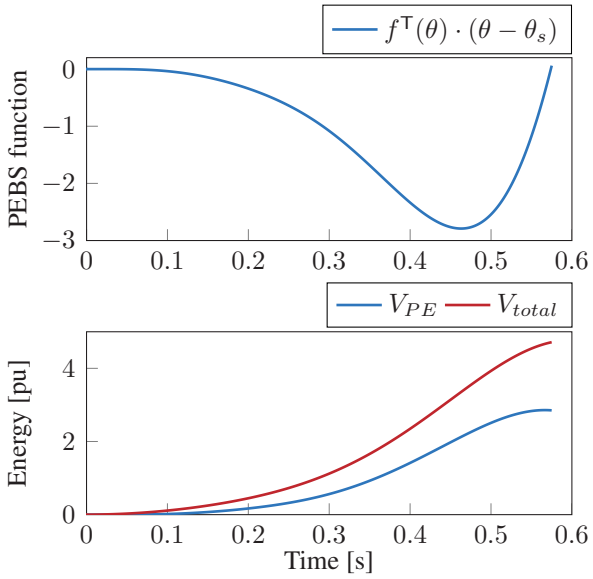
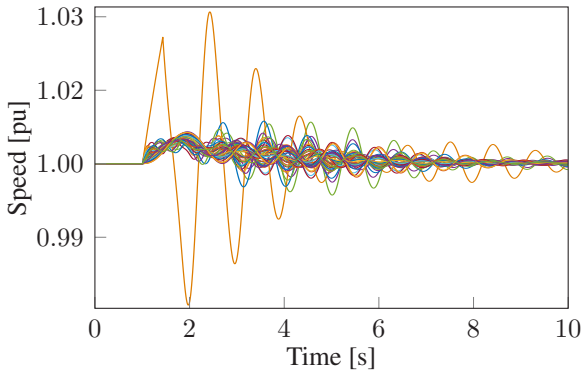


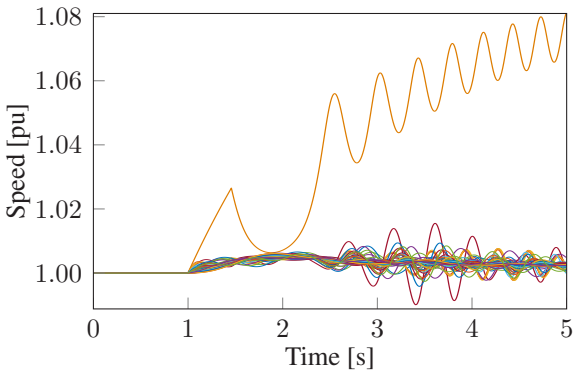
Figure 6.7: Monitoring PEBS function, total energy (V_{total}) and potential energy (V_{PE}) for a critical contingency

Figure 6.7 shows that the zero crossing of the potential energy directional derivative is reached at approximately $0.574\,\text{s}$, which corresponds of the point of exit of the stability region. The simulated contingency is therefore classified as a critical contingency. The maximum potential energy until the zero crossing point gives the estimate of the critical energy and is given as $V_{cr} = 2.858$ pu. It is worth noting that the zero crossing point in this case is reached at approximately the same time as the maximum potential energy, as observed in Figure 6.7. In this case, the developed method is in good agreement with the directional derivative approach and the maximum potential approach, but provides a less conservative estimate. In addition to classification of the contingency, the presented method is used to derive the required critical clearing time of the fault. From the definition in Section 6.2.4, the critical clearing time is estimated at the point when $V_{total} = V_{cr} = 2.858$ pu, giving $t_{cr} = 0.432\,\text{s}$.

In order to verify the classification in this case, the same contingency is analyzed using the time-domain simulation approach. The generator rotational speed response from the step-by-step simulation is shown in Figure 6.8a at the critical clearing time estimated from the direct method. The system shows a stable behavior for this clearing time. The exact critical clearing time in the time-domain simulation approach is determined by successively varying the fault clearing time until instability is observed in the post fault period. Figure 6.8b shows the rotational speed response at the time where unstable operation is first observed in the post-fault period. The critical clearing time in this case is $0.459\,\text{s}$. Compared to the direct method, the relative difference in the critical clearing time estimate is 5.88%, which shows a close agreement in the estimates from both approaches.



(a) Response for a fault duration equal to critical clearing time; i.e. $t_{cl} = t_{cr}(0.432\,\text{s})$.



(b) Response for a fault duration greater than critical clearing time; i.e. $t_{cl} > t_{cr}$.

Figure 6.8: Generator rotational speed response for a critical contingency using time-domain simulations

6.3.3 Performance and Reliability Evaluation

In addition to validating the accuracy of the method, it is important to evaluate the reliability and computational performance of the proposed assessment method. The evaluation is carried out in two ways. Firstly, the test scenario using the IEEE 118-bus network is used with fault clearing time of a three-phase short circuit fault for each contingency varied as follows: Case I – 2.0 s, Case II – 1.0 s, Case III – 0.5 s, Case IV – 0.2 s. Secondly, additional test networks are applied with varying complexity in terms of number of buses, generators, branches, and contingencies as defined in Table 6.1. The tests are carried out on an Intel(R) Core(TM) i7-8700 CPU @ 3.20 GHz, 32 GB memory system, running on 64-bit Microsoft Windows 10.

Performance

First, the runtime of the proposed method is evaluated using the four test scenarios defined for the IEEE 118-bus network with different fault clearing times. The total runtime of the main steps of the presented assessment algorithm is illustrated in Figure 6.9. As depicted in Figure 6.9, the time-domain simulation step to compute the system trajectory (TDS) and the contingency selection step ($Selection$) are the most computationally intensive steps of the simulation process. However, the runtime of these steps is observed to depend on the fault clearing time, i.e. the runtime decreases with fault clearing time as shown in Figure 6.9. This implies that the direct method achieves computational efficiency by limiting the intensive analysis stages to a very short time. On the other hand, the formulation of the admittance matrix ($Ybus$), the network reduction ($NetRed$) and the eigenvalue analysis (EV) steps are shown to be independent of the fault clearing time.

The runtime of the main steps in the assessment method is further evaluated using networks of varying complexity as described in Table 6.1. Figure 6.10 depicts the ratio of the runtime per contingency ($per\,Cont.$) of the main steps with respect to the total simulation time. It is observed from the figure that the time-domain simulation to derive the system trajectory is the dominant step for networks with a small number of generators. The time-domain simula-

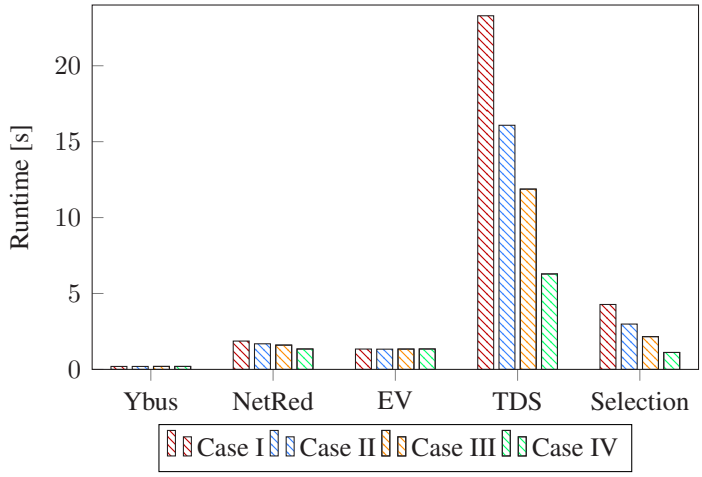


Figure 6.9: Performance of the main steps of the proposed method with varying critical clearing time

tion step is dominated by the numerical integration using the explicit Runge-Kutta integration method with linear complexity and the forward and backward network solution with quadratic complexity depending on the number of machines. As the network size increases, the eigenvalue analysis step becomes the dominant step with more than quadratic complexity with respect to the number of machines. The contingency selection step increases with polynomial complexity $O(m^c)$ for $2 \leq c \leq 3$, where m is the number of machines. Details of the total runtime for the tested cases are given in Appendix C.1.

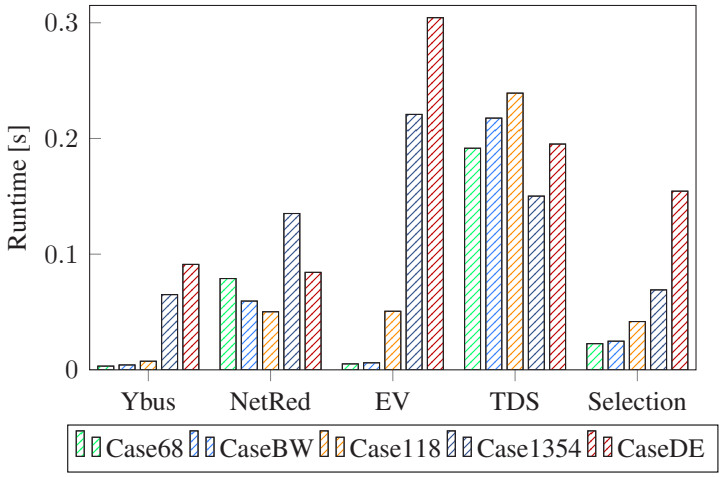


Figure 6.10: Ratio of runtime to the total simulation time per contingency of the main steps in the proposed method with varying network complexity

Reliability

The reliability in contingency selection is assessed in reference to the classification by the step-by-step time-domain simulation approach. Stability assessment in the time-domain simulation

is based on the angular separation between the interconnected machines in the system. A contingency is identified to be unstable if the rotor angular separation of any machine from the rest of the system is greater than $120°$, which is a common measure of the maximum angular separation of generators before loss of synchronism [96]. The four test cases of the IEEE 118-bus network with different fault clearing times are considered here. Table 6.6 shows the variation of the contingency selection results by the time-domain simulation approach and the selection accuracy of the presented direct method relative to the benchmark results of the time-domain simulation approach.

Table 6.6: Accuracy of the proposed direct method with respect to time-domain simulation approach

Test Case	Fault duration (s)	Time-domain selection		Direct method accuracy	
		Noncritical	Critical	Noncritical	Critical
Case I	2.0	135	353	90.37%	100%
Case II	1.0	144	344	97.92%	100%
Case III	0.5	164	324	90.85%	100%
Case IV	0.2	352	136	93.18%	100%

From the results presented in Table 6.6, it is observed that the direct method shows a high level of accuracy in classifying the network contingencies. The method identifies the unstable contingencies with 100% accuracy in all scenarios, which implies that all critical contingencies are identified as unstable. However, less accurancy is observed for the noncritical contingencies. This value represents the number of selected noncritical cases by the presented method compared to the actual stable (noncritical) cases defined by the time-domain simulation. The variation in the selection of stable or noncritical contingencies can be attributed to the conservativeness of the method, where the noncritical contingencies are falsely classified as critical (i.e. false critical cases). It should be noted that the above results do not include the contingencies identified as unclassifiable by the direct method.

Furthermore, the accuracy, reliability and computation speed of the proposed assessment method is compared to the state-of-the-art SIME-based method implemented as described in [112]. The test networks used for the comparison are described in Table 6.1. In each case, the fault clearing time is set to $200\,\text{ms}$ to measure the computational runtime. The assessment in the SIME-based method is carried out after $4\,\text{s}$ of the time-domain simulation, which is considered to be sufficient for capturing first swing instability as stated in [112]. Table 6.7 shows the comparison of the two methods with respect to benchmark time-domain simulations. As shown in Table 6.7, the proposed assessment method shows better accuracy and reliability than the SIME-based method in all the tested cases. In addition, the computational speed per contingency using the proposed method is observed to be 8 times faster for the smallest network and 2.5 times faster for the largest tested network than the SIME-based method.

In general, the presented method shows a desirable characteristic of very high probability of correctly classifying unstable contingencies as critical cases. Furthermore, the computational speed of the proposed method shows great potential in light of the requirement for fast stability assessment. However, the main question arising from the evaluation of the presented method is the notable conservativeness identified with the estimates. Estimations of the region of stability

Table 6.7: Comparison of the proposed direct method to the SIME-based method

Network	Proposed Method			SIME-based Method [112]		
	Accuracy (%)	Reliability (%)	Runtime per Cont.(s)	Accuracy (%)	Reliability (%)	Runtime per Cont. (s)
Case68	93.85 – 100	96.43 – 100	0.018	95.80 – 100	94.64 – 100	0.1435
CaseBW	95.85 – 99.75	96.63 – 100	0.0207	89.06 – 96.71	95.35 – 98.31	0.1482
Case118	90.37 – 97.92	100	0.0356	82.57 – 96.21	96.64 – 99.72	0.1774
Case1354	90.04 – 94.67	99.44 – 99.58	0.3216	89.97 – 94.31	98.64 – 99.57	0.9329
CaseDE	95.92 – 99.32	93.12 – 100	1.0018	90.32 – 93.08	90.63 – 93.48	2.5006

based on the controlling unstable equilibrium point (CUEP), such as the BCU (Boundary of stability region-based Controlling Unstable equilibrium point) method are considered to provide less conservative assessment results. Application of such improved methodologies, especially in the assessment of larger networks, is to be taken into consideration in future work.

6.4 Summary

The current chapter presents an alternative method for the assessment of transient stability in power systems. The analysis approach is based on a Lyapunov direct method using the PEBS criterion for fast assessment of stability. The novelty in the presented method is the derivation of a new approach combining the directional derivative of the potential energy function and the maximum potential energy criterion in order to reduce the conservativeness in the estimation of the stability region in the analysis process. It is shown from the evaluation results that the presented method is able to reliably and accurately screen out noncritical contingencies from the large number of network contingencies. This method therefore presents a solution to address the need for continuous analysis in system stability studies in order to account for the constantly changing operating conditions in the current power system environment. The remaining objective is how to achieve fast as well as detailed analysis in power system studies. This can be achieved by combining the advantages of the presented direct method and the parallel time-domain simulation approach. This objective is addressed in the next chapter through the development of a hybrid analysis approach.

7 Hybrid Analysis Method and System Integration

The computational methods described in the previous chapters address separate requirements in the stability analysis process for time-domain simulations and a direct analysis method. A hybrid analysis method is presented in the current chapter in order to combine the computational advantages of the individual methods. The resulting benefit of the combined method is a computational analysis approach for fast and detailed analysis of power system transient stability. Furthermore, the present chapter describes the integration of the computational methods developed in the present thesis into an interactive network modeling and visualization framework, *e*ASiMOV [114, 115]. Currently, the *e*ASiMOV framework supports steady state analysis of power systems. Integration of the dynamic simulation methods therefore extends the framework to transient stability and contingency analysis. Simulation, analysis, and visualization in the integrated framework are presented to illustrate the added features in the software framework contributing to the Energy Lab 2.0 testing facility.

7.1 Hybrid Analysis Method

The need for continuous transient stability analysis necessitates a balance between analysis efficiency and simulation accuracy in the power system computational methods. The current section presents a method to address the above requirement. From the analysis accuracy point of view, the time domain simulation method presented in Chapter 3 and extended to parallel computation in Chapter 5 shows reasonable accuracy in the analysis results. On the other hand, the direct analysis method presented in Chapter 6 reduces the computational complexity by limiting the computationally demanding numerical integration to a very short period of the simulation. Therefore, the ability of the direct method to provide a fast approach for determining transient stability addresses the requirement of computational efficiency in the analysis process. Analyzing the benefits of the two approaches shows that a natural way of achieving the essential continuous power system analysis, while maintaining accuracy and details in the analysis, is through coupling the direct-method and time-domain simulations. The resulting approach is referred to as a hybrid analysis method in the present thesis.

The current chapter is based on the work published in [126, 164].

M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Dynamic Simulation Module for Power System Transients Analysis in the *e*ASiMOV Framework," in *2017 European Modelling Symposium (EMS)*, Manchester, UK, pp. 157–162, November 2017.

M. Kyesswa, H. K. Çakmak, L. Gröll, U. Kühnapfel, and V. Hagenmeyer, "A Hybrid Analysis Approach for Transient Stability Assessment in Power Systems," in *2019 IEEE Milan PowerTech*, Milan, Italy, pp. 1–6, June 2019.

7.1.1 Method Structure

Formulation of the hybrid analysis method is based on the fundamental characteristics of the direct-method and time-domain simulations. Figure 7.1 shows the components of the hybrid analysis method. Considering the computational merits in each of the methods, the hybrid analysis approach is developed as a two-stage process. The first stage of the analysis is a fast classification stage. This is followed by the detailed analysis of the system response in the second stage.

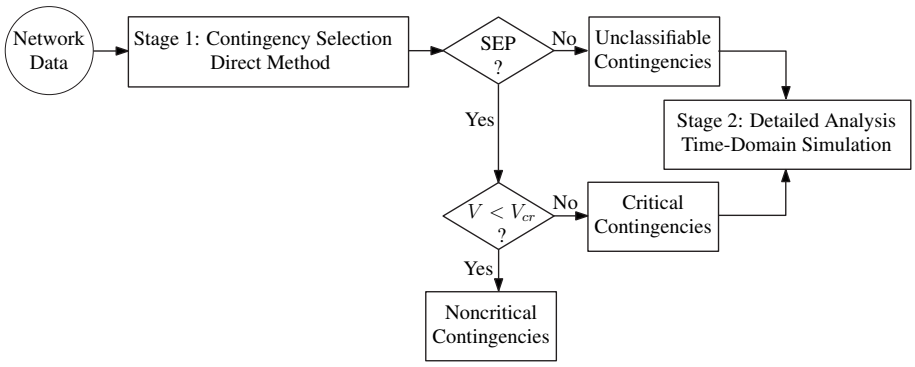


Figure 7.1: Overview of the hybrid method workflow

In the first stage, the direct method is applied for the classification process. The objective at this stage is to separate network contingencies into three groups as shown in Figure 7.1: unclassifiable, noncritical, and critical contingencies. As described in Chapter 6, the summary of the selection categories is as follows: Unclassifiable contingencies are those where the system state of stability cannot be concluded by the direct method if the post-fault stable equilibrium point (SEP) does not exist; Noncritical contingencies result from network disturbances where the system remains stable for the set simulation duration, i.e. the total energy gain during fault (V) is less that the critical energy (V_{cr}); and Critical contingencies as those resulting from disturbances which drive the system into a state of instability before the end of the set simulation time, i.e. $V > V_{cr}$. In other words, the first stage is a screening process where the noncritical contingencies are eliminated from further analysis in the combined analysis process. Therefore, the selection stage of the analysis reduces the number of contingencies for detailed analysis to only the potentially dangerous network disturbances.

The second stage of the method addresses the need to derive the details of system variables and visualize the actual system response subject to specific disturbances. For this, the time domain simulation approach is applied for further analysis of the selected unclassifiable and critical contingencies using detailed component modeling and step-by-step numerical integration methods. This stage therefore provides insight into the system behavior during the disturbance and in the post-fault state. In addition, the observed response is used to define the criticality of power system components with respect to the location of a disturbance.

An additional benefit of the coupled analysis approach is that the internal numerical integration methods in the time-domain simulation are directly applied at the initial stage of the direct method involving numerical integration to determine the system trajectory during the fault pe-

riod. This additionally contributes to the computational speed at the selection stage since the system state variables are directly analyzed during computation of the fault-on trajectory.

7.1.2 Hybrid Analysis Process

General Input Data

The hybrid method is implemented in the Matlab environment by combining the direct-method and time-domain simulation algorithms. The simulation process starts with the definition of the network structure, the system dynamic parameters and list of contingencies to be analyzed. As in any dynamic simulation process, the system is first initialized to a steady state. A single power flow computation based on the Matpower Matlab package is executed for the initialization at the beginning of the simulation in order to determine the pre-fault steady state network conditions. During contingency analysis, the pre-fault network conditions are assumed to be common for all the simulation cases to be analyzed. This implies that the starting conditions are similar for all the analyzed contingencies in the given network.

Contingency Selection Stage

The inputs for the selection stage of the hybrid method are the pre-fault steady state conditions, list of contingencies – in terms of fault type, fault location and the fault clearing mode – and the simulation duration. The duration of the computation is set according to the maximum time required for simulation of the fault on the network. This duration corresponds to the minimum reaction time of the protection devices in response to the disturbance in practical applications. The objective of the simulation at this stage is to identify and quantify contingencies that result in system instability before isolation of the disturbance from the network.

The first of the three contingency categories is identified during the preselection step of the selection stage. This step eliminates the unclassifiable cases, as described in Section 6.2.3, where stability cannot be analyzed using the direct method if the post-fault equilibrium state does not exist. Such contingencies can only be analyzed using the time-domain simulation approach and are therefore directly sent to the second stage of the process. The selection of the remaining contingencies is based on comparison of the transient energy gained during the fault period and a critical energy value as shown in Figure 7.1. Contingencies resulting in a gain in transient energy greater than the critical energy are categorized as critical contingencies. These are added to a list of contingencies requiring further detailed analysis in the second stage. Noncritical contingencies, where the transient energy during fault is less than the critical energy for the whole simulation duration, are eliminated from further analysis in the second stage.

During the contingency selection process, the following results are saved to a file: eigenvalues for each contingency; the contingency selection results including the corresponding critical energy (V_{cr}) and critical clearing time (t_{cct}); and profiles of the transient energy, potential energy, and the derivative of the potential energy in form of graphical plots. The critical energy and critical clearing time are necessary for ranking the critical contingencies.

Detailed Analysis Stage

At the second stage of the hybrid analysis process, the identified unclassifiable contingencies and critical contingencies are read into the time-domain simulation module for further analysis. The input data for analysis of each contingency are the fault location, line switching action, and the corresponding critical clearing time estimate from the first stage. The critical clearing time defines the fault duration in the time-domain simulation for each contingency and the total simulation time is set in a way that allows a clear analysis of the post fault system response. For the analysis in the present thesis, the simulation time is set to 10 seconds.

The simulation results of the main system variables – which include the generator rotor angle, rotational speed, and bus voltages – are saved at this stage. System stability is analyzed from the derived responses, whereby the system is stable if the corresponding responses settle to a steady state operating point in the post-fault state. Otherwise, the system is defined as unstable

Algorithm 7 : Overview of the simulation workflow in the hybrid analysis method

Inputs: Network structure data
Initialize system to steady state
stage 1:
for *each node* n **do**
  trigger fault simulation
  for *each branch* b *connected to node* n **do**
    open branch and compute SEP
    if *SEP does not exist* **then**
      save as unclassifiable contingency
    else if *SEP exists* **then**
      while $t < fault\ time$ **do**
        determine θ_t and $\tilde{\omega}_t$
        determine $V_{total}, V_{PE}, f^{\mathsf{T}}(\theta_t) \cdot (\theta_t - \theta_s)$
        if $f^{\mathsf{T}}(\theta_t) \cdot (\theta_t - \theta_s) = 0$ **then**
          determine V_{cr}, t_{cct}
          save as critical contingency
        else if $f^{\mathsf{T}}(\theta_t) \cdot (\theta_t - \theta_s)$ **then**
          save as noncritical contingency
Return: List of selected contingencies c in terms of n, b, t_{cct}
stage 2:
Inputs: read in list of selected contingencies c
for *each contingency in* c **do**
  while $t < simulation\ time$ **do**
    fault settings: bus = n; duration = t_{cct}; line switch = b
    run time-domain simulation
    compute detailed system response
Return: State variables at each time step

if the system responses are diverging or oscillating. Based on the detailed results, decisions for corrective action or network reconfiguration can be made to ensure network stability. In time-domain simulations, the fault clearing time is successively varied in order to determine the exact critical clearing time. Additional information about the importance of network components can be derived from the analysis of the unclassifiable contingencies in the second stage. For such contingencies, the time-domain simulation approach starts with a clearing time of one time step. If the response is unstable for a fault duration of one time step, it is concluded that the post fault stable state does not exist for the analyzed contingency. The components involved in the switching action are considered to be very critical to system stability. Algorithm 7 shows the pseudo code of the work flow in the hybrid simulation framework.

7.1.3 Performance Evaluation

To analyze the performance of the hybrid method, different network structures of varying complexity in terms of number of generators and buses are applied as described in Table 6.1. The goal is to determine the total time required for the assessment of the whole list of contingencies using the hybrid method in comparison to assessment using time-domain simulations based on the parallel approach described in Chapter 5. For purposes of illustration, the fault clearing time is set to $500\,\text{ms}$ for each tested contingency. As described in Section 7.1.2, the hybrid method executes the first stage, i.e. direct method ($DM\ stage$), for each contingency while the second stage based on detailed time-domain simulations ($TDS\ stage$) is limited to the identified critical cases. The total runtime of the hybrid method ($Total$) is the sum of the duration of the two stages. On the other hand, the calculated runtime for the time-domain simulation ($TDS\,method\,runtime$) is the total time for the analysis of the whole list of contingencies. The time-domain simulation is run for $10\,\text{s}$ at a step size of $1\,\text{ms}$ for each contingency.

Details of the workstation used for the tests are as follows: Intel(R) Core(TM) i7-8700 CPU @ 3.20 GHz, 32 GB memory system, running on 64-bit Microsoft Windows 10. Table 7.1 summarizes the simulation runtime using the two approaches for the different test networks. The runtime reduction in Table 7.1 represents the percentage time saved by applying the hybrid method for the complete assessment compared to using a pure time-domain simulation.

Table 7.1: Evaluation of simulation runtime of the hybrid method in comparison to time-domain simulations for the assessment process

Network	Contingencies		TDS Method	Hybrid Method Runtime (s)			Runtime reduction (%)
	Total	Critical	Runtime (s)	DM stage	TDS stage	Total	
Case68	186	65	321.40	3.37	114.27	117.64	63.40
CaseBW	551	136	1370.96	11.39	443.57	454.96	66.81
Case118	490	162	1256.25	17.46	416.48	433.96	65.46
Case1354	5336	1082	89086.84	1716.07	25145.57	26861.64	69.85
CaseDE	6971	892	151915.6	6983.4	29579.44	36562.84	75.93

The results in Table 7.1 show that the hybrid method attains a significant reduction in the total time required for the assessment process compared to applying pure time-domain simulations.

It should also be noted that the number of selected critical contingencies for analysis at the second stage of the hybrid method depends on the considered fault clearing time. In this case, a clearing time of 500 ms is applied for testing purposes, which explains the large number of critical cases. However, shorter response times (fault clearing times) as low as 50 ms (2.5 cycles) are considered in practice, for example in high voltage transmission systems. As shown in Table 6.6, critical cases decrease as the clearing time decreases. This corresponds to a reduction in cases requiring detailed analysis at the second stage, and thus reducing the total runtime of the hybrid method. Furthermore, since the analysis of each contingency is independent from the other contingencies, the overall runtime of the assessment process can be reduced through distribution of tasks to several processors to perform the assessment in parallel.

7.2 Integration into the Energy Lab 2.0 Platform

The current section presents the integration of the developed dynamic computational methods into a software framework which contributes to the Energy Lab 2.0 testing facility [14]. The Energy Lab 2.0 is intended for studying the interaction of components in future energy systems as well as testing new technological solutions necessary for the energy transition. A central element in the Energy Lab 2.0 is the smart energies system simulation and control center, which is made up of three components: a power-hardware-in-the-loop experimental field; an energy simulation and analysis laboratory; and a control, monitoring and visualization center [14]. The computational methods developed in the present thesis contribute to the energy simulation and analysis laboratory component of the system. Integration of the developed computational methods into the Energy Lab 2.0 facility is achieved through the *e*ASiMOV (for energy system Analysis, Simulation, Modeling, Optimization and Visualization) software framework as a contributing package to the simulation laboratory. In the current state, the *e*ASiMOV software framework supports power flow simulations based on the Matpower package. The main objective of integrating the computational methods under the hybrid software framework is to extend dynamic simulations to the *e*ASiMOV framework, and thus apply the methods in the Energy Lab 2.0 testing environment.

7.2.1 Overview of the *e*ASiMOV Software Framework

The *e*ASiMOV software framework – comprising of energy system analysis, simulation, modeling, optimization and visualization components – provides an extendable framework for interactive modeling and analysis of energy systems [114, 115]. The goal of the *e*ASiMOV software framework is to develop a one-user interface, one-model and multiple-simulator policy with support for model exchange and computational scalability. It therefore introduces a distributed system architecture with a collection of software tools. The main components of the *e*ASiMOV distributed architecture are the modeling component – *e*PowMod, the simulation module – *e*PowSim, and the visualization module – *e*PowVis. A description of the software modules comprising the *e*ASiMOV framework is given in Appendix C.2.

In the current state of the framework, the *e*PowSim module provides steady state analysis of the power grid using power flow computations. The computational modules of the hybrid simu-

lation method are integrated into the *e*ASiMOV framework as part of the *e*PowSim – simulation module. Figure 7.2 shows the user interface of the *e*ASiMOV framework modeling interface, with the new computational methods integrated under the *e*PowSim module as shown in the figure.

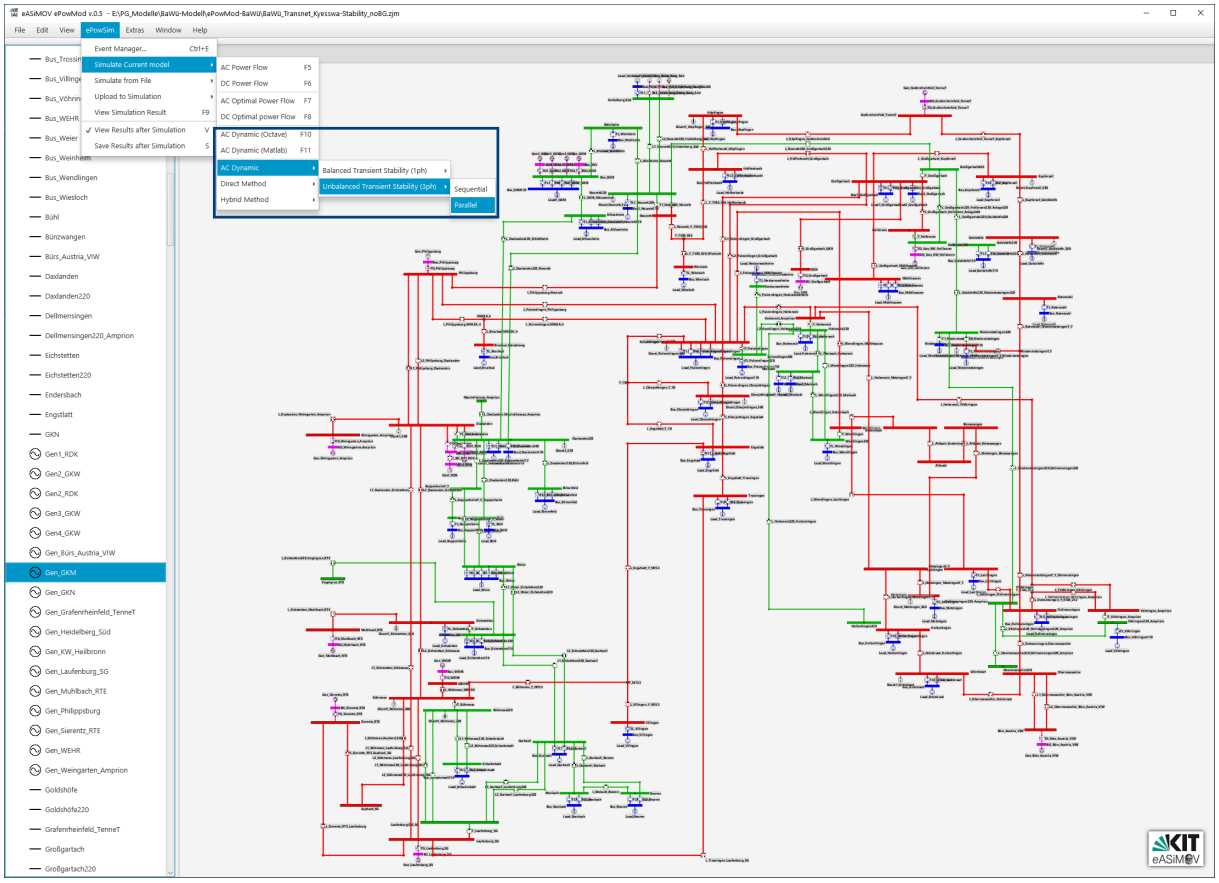


Figure 7.2: Interactive user interface in *e*PowMod and integrated dynamic simulation modules in the *e*PwoSim module

7.2.2 Dynamics Simulation in the *e*ASiMOV Framework

The dynamics simulation package is integrated into the *e*ASIMOV framework as a designated part of the *e*PowSim simulation module. In general, the analysis process starts with the interactive network modeling, which is performed in the *e*PowMod module. The modeling component then generates a Matpower file representing the network model as the input to the dynamic simulation module. This step applies to both power flow analysis and dynamics simulation. Specific to the dynamics simulation, an additional input file is defined for the dynamic system parameters as described in Section 3.1.1.

A flexible choice of the basic computing environment is provided between Matlab and Octave[1] in the integrated framework. This option is included in the interactive front-end modeling user interface shown in Figure 7.2. In the Matlab computing environment, the modeling module generates a Java process that passes the input files and the simulation settings – including the simulation duration and integration step size – to the dynamics simulation module. On the other hand, the Octave computing environment is coupled directly via the JavaOctave library[2]. During the dynamic simulation, the simulation results are saved by updating the respective variables in the Matpower case file at every time step. The updated variables include the voltage magnitude and angle, and the active and reactive power of the generators. The grid frequency is estimated using the dynamic frequency approximation as described in Section 4.3.1. Bus frequency approximations are inserted into the result file to form a new data structure. The results are defined specifically in form of Matpower files to maintain a format compatible with the *e*PowVis visualization module. As part of the interactive analysis of the simulation results, the visualization module loads the saved result files and displays the results in form of heat maps at every time step.

7.3 Analysis in the Integrated Framework

The present section describes the analysis procedure in the hybrid method coupled with the visualization of simulation results in the *e*ASiMOV framework. The main aim of the presented results is to illustrate the functionality of the integrated software framework. For the test simulations presented in the current section, the modified Baden-Württemberg network described in Appendix A.2 is applied as the benchmark model. In the following, selected cases from the unclassifiable and critical contingencies are presented to illustrate the analysis and visualization in the second stage using the integrated framework.

7.3.1 Unclassifiable Contingency

One case of an unclassifiable contingency is a fault on bus 131, cleared by switching out the line between bus 12 and bus 131, i.e. fault bus 131 and line switch 12–131. It is identified to be unclassifiable according to the analysis of the eigenvalues of the state matrix at the post-fault equilibrium point as shown in Figure 7.3. In this case, the state of stability following such a contingency can only be derived in the second stage using time-domain analysis. Figure 7.4 shows the response of the generator frequency for a fault duration of 10 ms in stand-alone mode of the hybrid method. An alternative representation of the frequency response is shown in the visualization module of the integrated framework as depicted in Figure 7.5 for the critical section of the network.

The analysis in the time-domain simulation therefore concludes that the switching case always results in unstable operation of the network in the post-fault period irrespective of the fault duration. The responses depicted in Figures 7.4 and 7.5 can be explained as follows: From the network structure, bus 131 is connected to a generator bus at bus 12. Switching out line

[1] https://www.gnu.org/software/octave/
[2] https://github.com/prateek/javaoctave

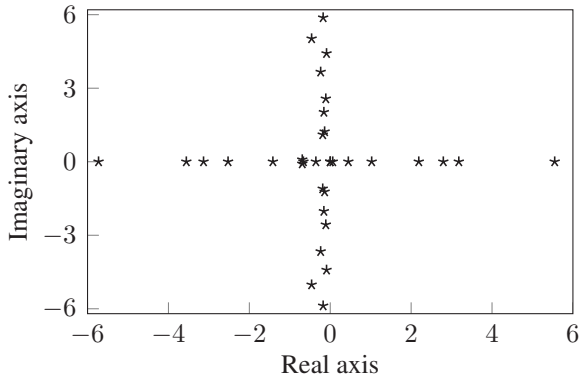


Figure 7.3: Eigenvalues analysis of state matrix for post-fault system without line 12–131. Existence of positive real eigenvalues indicates an unstable equilibrium point in the post-fault state.

12–131 to clear a fault on bus 131 results in disconnection of a generating unit from the system. This causes the immediate acceleration of the disconnected generator, due to the imbalance between the mechanical power (P_m = maximum) and the electrical power ($P_e = 0$) seen by the corresponding generator. Interactive visualization results of the whole network in the *e*PowVis module are given in Appendix C.2.

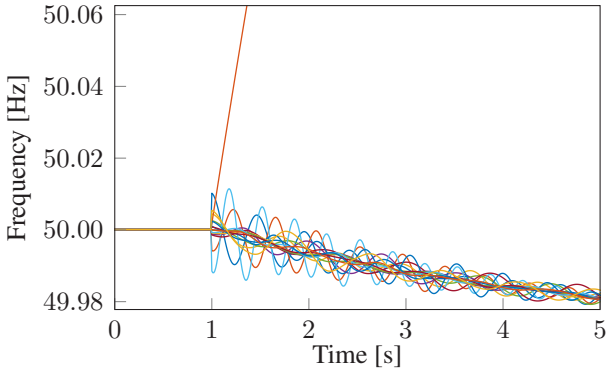
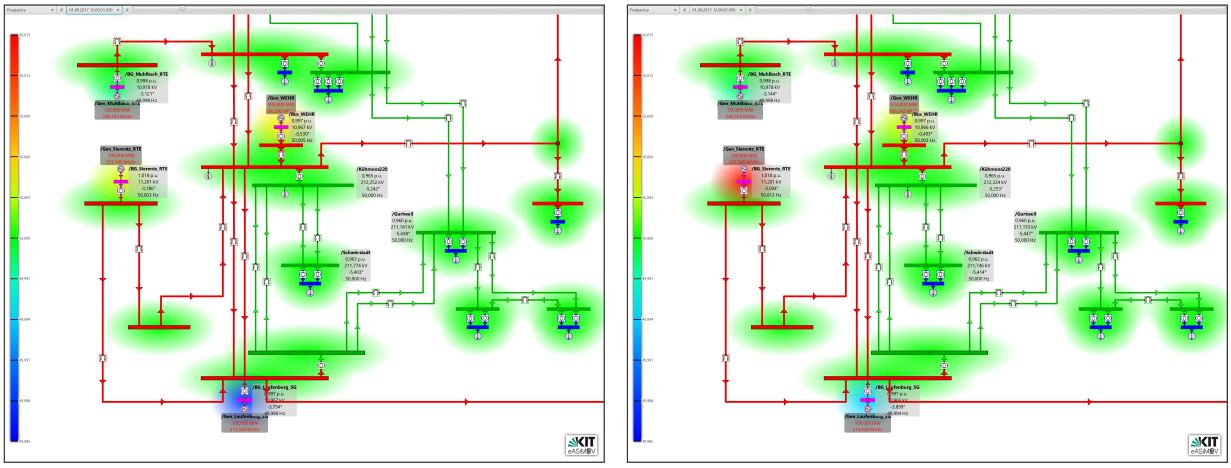


Figure 7.4: Response of generator frequency for a fault duration of 10 ms showing one generator running out of synchronism with the rest of the generators in the post-fault state.

7.3.2 Critical Contingency

An example of a critical case is as a fault on bus 95, cleared by a switching out the line between bus 95 and 90, i.e. fault bus 95 and line switch 95–90. Figure 7.6 shows how the hybrid method classifies the case as a critical contingency using the potential energy function and monitoring of the transient energy in the first stage.

The time-domain response at the critical point is shown in Figure 7.7, which depicts the generator frequency in the second stage of the simulation method for the critical contingency. Such analysis provides more insight to the system operator regarding the actual state of the



(a) Visualization of system frequency at $t = 1.030\,\mathrm{s}$ (b) Visualization of system frequency at $t = 1.090\,\mathrm{s}$

Figure 7.5: Interactive visualization of the response of system frequency in the *e*ASiMOV framework for the critical part of the network. Depicted frequency scale ranges from $49.985\,\mathrm{Hz}$ to $50.015\,\mathrm{Hz}$.

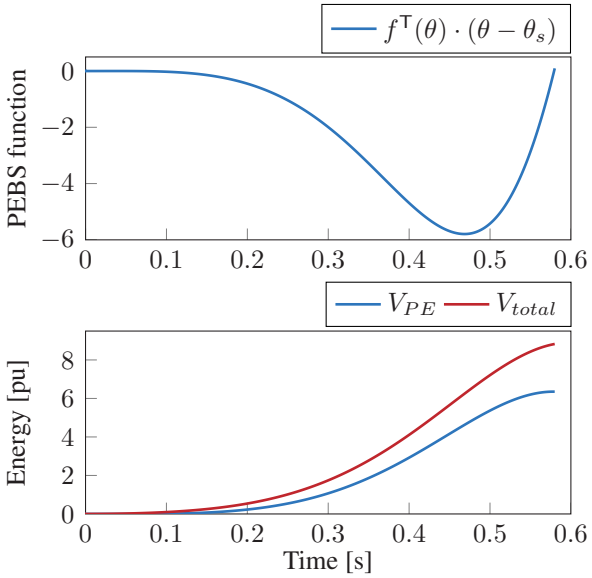


Figure 7.6: Monitoring PEBS function, total energy and potential energy for a critical contingency. Estimated critical clearing time at a point where $V_{total} = max(V_{PE})$ is $0.471\,\mathrm{s}$.

machines. Furthermore, the corresponding response in the interactive visualization module of the *e*ASiMOV framework is shown in Figure 7.8. Beyond the critical clearing time, the response of the generator frequency is shown in Figure 7.9 for the time-domain representation and in Figure 7.10 using the visualization module. Details of interactive visualization in the whole network are given in Appendix C.2.

From the results described above, it is shown that the combined analysis in the hybrid method is able to identify critical contingencies for further analysis using time-domain simulations which provide detailed information about the actual response of the system to network dis-

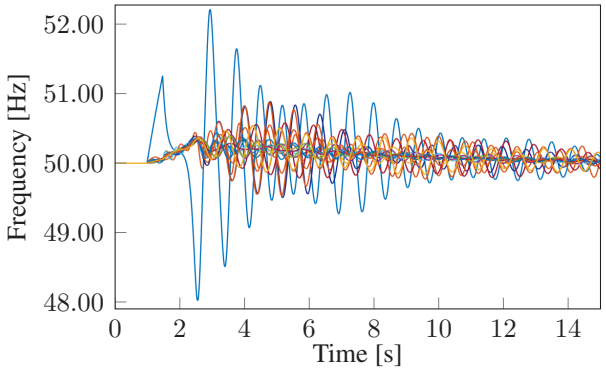
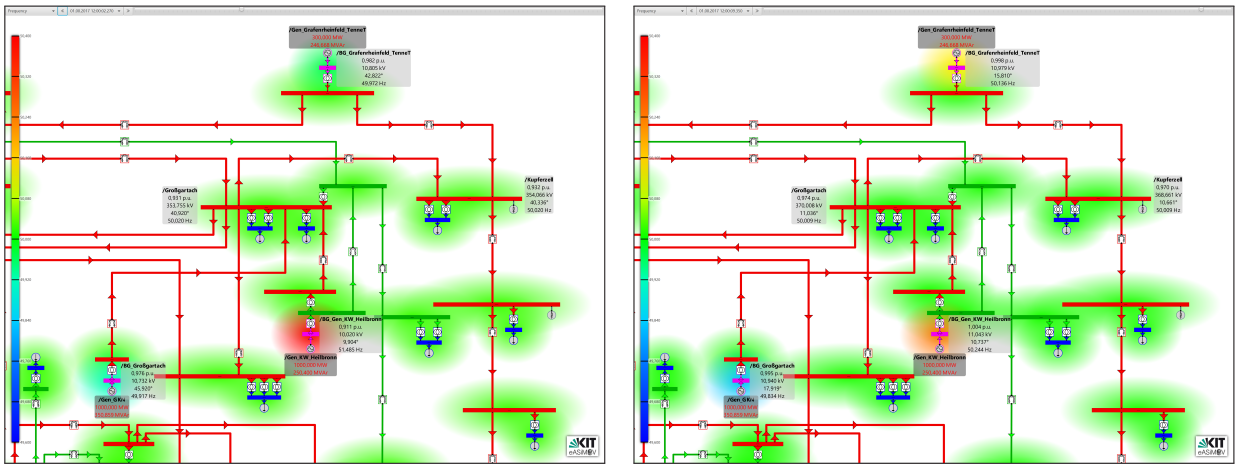


Figure 7.7: Response of generator frequency for a fault duration $t_{cl} = 0.471$ s less or equal to the critical clearing time (i.e. $t_{cl} \leq t_{cr}$). Generators return to synchronous operation in the post-fault state.



(a) Visualization of system frequency at $t = 2.270$ s

(b) Visualization of system frequency at $t = 9.350$ s

Figure 7.8: Interactive visualization of the response of system frequency in the *e*ASiMOV framework for a fault cleared before the critical clearing point. Critical network section is shown with frequency scale ranging between 49.6 Hz and 50.4 Hz.

turbances. The two analysis stages are also shown to be complementary in the hybrid method. Therefore, the combined analysis forms a framework for fast contingency selection and detailed system transient analysis. Furthermore, the integration of the hybrid method into a modeling and visualization framework shows an additional benefit in the analysis process by providing interactive visualization of the system response.

7.4 Summary

The contributions in the current chapter are summarized as follows: Firstly, a new hybrid method is developed which addresses the open question regarding the need for a method providing fast and detailed stability analysis for the continuous assessment of power system transients. Secondly, the hybrid method is integrated in a framework which facilitates testing of the method

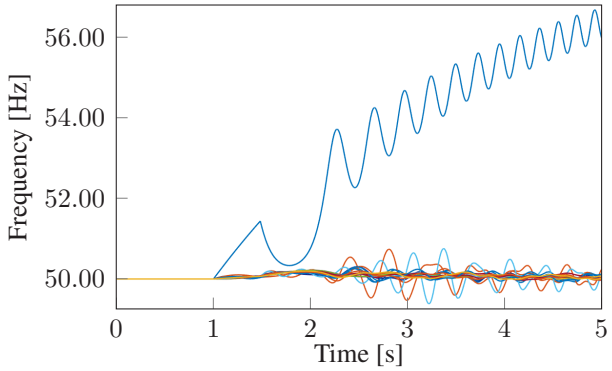
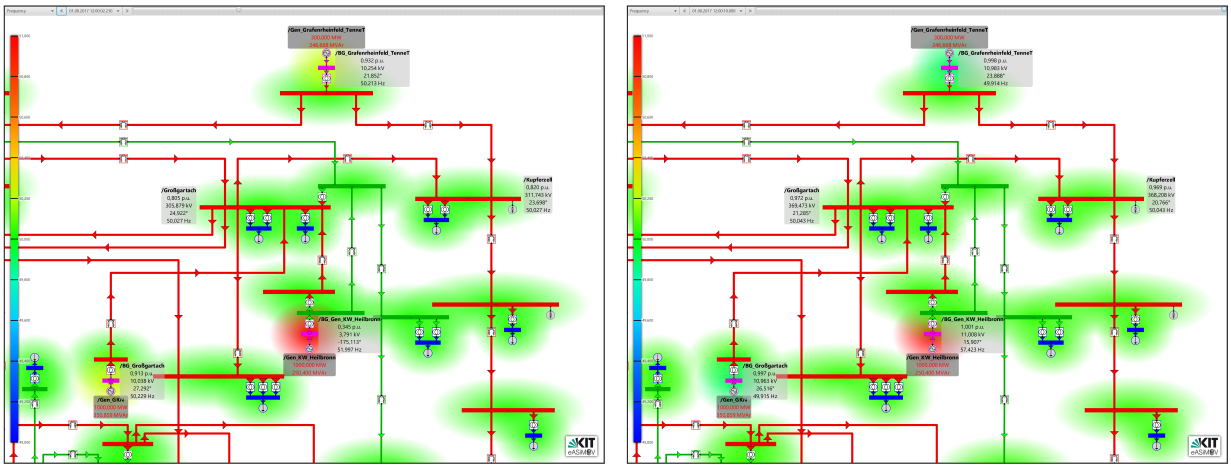


Figure 7.9: Response of generator frequency for a fault duration $t_{cl} = 0.483\,\text{s}$ greater than the critical clearing time (i.e. $t_{cl} > t_{cr}$). One generator runs out of synchronism with the rest of the generators in the post-fault period.



(a) Visualization of system frequency at $t = 2.230\,\text{s}$

(b) Visualization of system frequency at $t = 10.00\,\text{s}$

Figure 7.10: Interactive visualization of the response of system frequency in the *e*ASiMOV framework for a fault cleared at a time beyond critical clearing time. Critical network section is shown with frequency scale ranging between 49.0 Hz and 51.0 Hz.

and the included component models in a smart grid testing environment. The framework with the integrated dynamics simulation module is provided with a graphical user interface, an interactive network editor and a visualization module, which give it an advantage over the existing research based simulation tools. Nonetheless, for the integrated framework to be applied for online dynamic security assessment in the smart grid context, it should be combined with real-time measurements and state estimation procedures. This enables continuous analysis of system stability based on the current state of the network. Future related work should take this measure into consideration.

8 Conclusion and Outlook

Major changes are seen in the current power system operating environment mainly due to increasing network sizes, increasing integration of renewable energy source, and the growing electricity demand. These changes in the structure, nature, and complexity of the power system directly impose additional challenges and requirements in the stability analysis process. One of the requirements is to increase model complexity in order to accurately represent the current state of the power system. This in turn results in an increase in complexity of the problem under analysis. Additionally, the changes in the context of the energy transition result in a continuously changing operating point. Therefore, the computational methods used in stability analysis are required to evolve with the changing power system analysis problem in order to cope with the introduced complexity and the requirement for continuous stability analysis during network operation. The current thesis addresses some of the requirements in the stability analysis process by developing models and computational methods which are combined into a new parallel hybrid analysis method.

Initially, new models are developed for the representation of the conventional power system under balanced and unbalanced conditions. Corresponding simulation functions are developed for the analysis of power system transients, where a new analysis method for combining symmetrical and asymmetrical transients' analysis is developed. In the context of the energy transition, models of wind and PV generation systems are developed to integrate the analysis of renewable energy sources in system stability studies using time-domain simulations. This is followed by parallelization of the analysis methods in order to address the introduced complexity in the analysis process using time-domain simulations. For this, a new parallel time-domain simulation method is developed based on graph partitioning and a parallel-in-space approach. The requirement of fast stability analysis is addressed by developing a direct method with a characteristic of fast stability assessment without explicit numerical integration and computation of the controlling unstable equilibrium points. Furthermore, the developed methods are combined into a hybrid analysis approach for fast and accurate stability assessment which can be applied for continuous analysis of system stability during network operation. Finally, the developed methods as well as component models are integrated into an energy system analysis framework that provides an interface to a smart grid testing environment. The main contributions in the current thesis are summarized as follows:

- New models for the representation of the conventional power system under symmetrical and asymmetrical operating conditions. Applying the developed models in system stability studies eliminates the limitation of the analysis to only three-phase balanced networks and therefore allows the representation of the true state in practical power systems.

- Development of a new flexible, extendable and scalable research grade computational method for time-domain simulations. The method is developed for the numerical solu-

tion of the power system equations based on the derived mathematical representation. Therefore, the method is used for the analysis of balanced and unbalanced power system network transients based on the steady state assumption and symmetrical components technique. The mathematical models and derived solution methods are used to develop a Matlab-based toolbox.

- Dynamic modeling of renewable energy sources – solar photo-voltaic and wind power – based on network standards for the interfacing converter-connected generation sources to the grid. For this, new models are developed to represent the inverter functionality as the interfacing point of the renewable energy sources to the grid. The dynamic models of the renewable energy components are integrated in the time-domain simulation toolbox.

- Development of a new parallel time-domain simulation method based on a new extended graph partitioning method and a parallel-in-space solution approach using a series of efficient direct solvers. This addresses the computational complexity involved in time-domain simulations through the use of high performance computing technology. The method is developed in Julia, a high performance programming language.

- A direct stability analysis approach for deriving the stability region using the potential energy boundary surface (PEBS) method with reduced conservativeness and without the computationally expensive task of determining the controlling unstable equilibrium points. The developed direct method provides a fast stability assessment technique.

- A new parallel hybrid analysis method for fast and detailed system stability analysis based on the coupled direct analysis method and parallel time-domain simulation method. The hybrid method enables continuous analysis of a whole set of contingencies and detailed analysis during network operation.

- Integration of the developed methods into the *e*ASiMOV simulation platform to create a framework for interactive modeling, editing, graphical representation and continuous interactive analysis and visualization of the dynamic behavior in power system networks. The integrated framework provides an interface to the Energy Lab 2.0 platform.

The contributions highlighted above show that the current thesis fulfills the intended objectives regarding modeling and development of computational methods. However, a number of open questions should be addressed in future work to extend the analysis methods in order to address the continuing transformation of the power system due to the energy transition. The mathematical models derived in the present thesis focus mainly on the generation side of the power system. However, several components such as HVDC, FACTS devices, energy storage systems are becoming increasing applied in power system. Models for the analysis of system stability should therefore include detailed modeling of such components. Since HVDC and FACTS devices are fast acting devices with electronic switching and time responses in the electromagnetic range, it is also necessary to extend the numerical solutions to account for electromagnetic transients. In addition, future work should consider further validation of the

high-level control functions for interfacing the renewable energy generators based on measured data from real plants.

As described in Section 5.4.3, the speedup in the parallel time-domain simulation method is affected by the communication overhead between the processes. Future work should consider optimization of the parallel method to reduce the data exchange overhead and improve the computational speedup. One possibility of achieving this could be implementing the parallel algorithm in a version of the Julia programming environment (v1.3.0) with better support for multi-threading of nested loops and provides general task parallelism properties, as proposed in [171]. In addition, since the solution in the time-domain simulation is a memory bound problem, another possibility to improve the speedup is to use GPU computing to take advantage of the larger memory bandwidth within graphic cards compared to CPUs. In terms of the solution technique, the applied BBDF technique should further be compared to the Multi-Area Thévenin Equivalent (MATE) algorithm in order to analyze possibilities of computational benefits in using a branch splitting technique for the parallel solution formulation.

The accuracy at the selection stage of the hybrid method could also be improved by applying state-of-the-art approaches which use the Boundary of stability region-based Controlling Unstable equilibrium point (BCU) method for defining the stability region. However, since the BCU method includes the expensive task of computing the controlling unstable equilibrium point, further work should address the additional complexity if the BCU method is applied in the hybrid analysis process. In addition, higher order models and new components such as HVDC and FACTS devices should be considered in the formulation of the transient energy function.

Finally, as the hybrid analysis method is aimed towards application in online stability analysis, the integrated framework should be combined with real-time measurements and state estimation procedures in the smart grid testing environment. As an example application case for this setup is the Karlsruhe Institute of Technology Campus North network with installed smart meters in substations. Based on the IEC 61850 protocol, the measured data can be sent to a real-time simulated network of the campus, which is then used to define the actual state of the network during the online stability assessment. The Energy Lab 2.0 testing facility of the Karlsruhe Institute of Technology provides a possible framework for achieving such a setup.

A Test Systems

A.1 IEEE 9-Bus Test Network

The IEEE 9-bus network consists of three generators (G1, G2, and G3), nine buses (B1 to B9) and three loads (L1, L2, and L3). T1, T2, and T3 represent transformers as part of the branch elements in addition to the transmission lines as shown in Figure A.1. The parameters of the transmission lines, loads, power flow data and the dynamic parameters of the generators are adapted from [18].

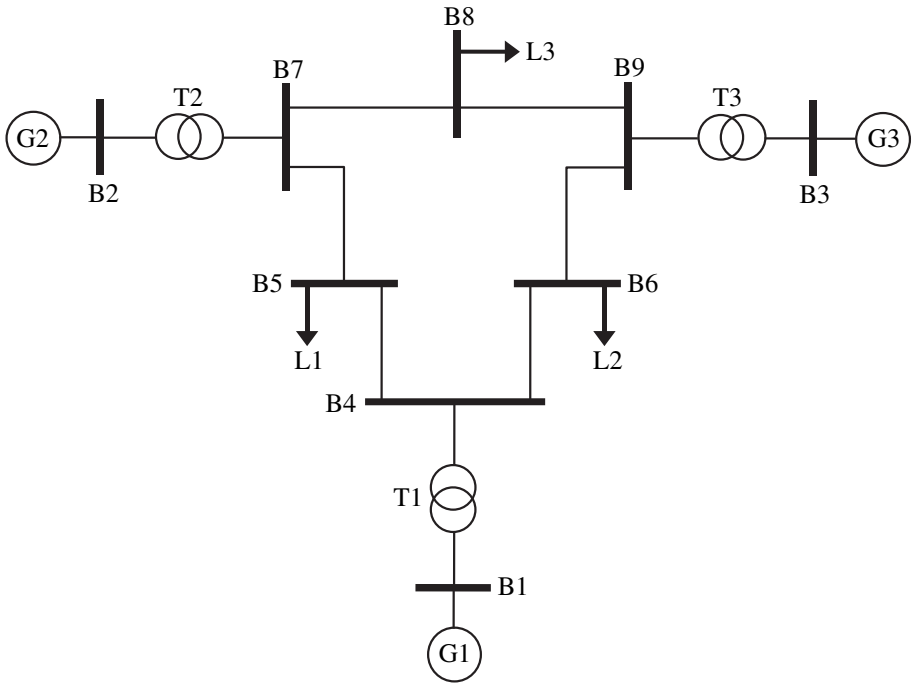


Figure A.1: IEEE 9-Bus network structure

A.2 Baden-Württemberg Transmission Network

The structural representation of the Baden-Württemberg transmission network used for testing the developed renewable energy generation systems presented in Chapter 4 is illustrated in Figure A.2. The network consists of 17 generators, 149 buses, 225 branches and 49 loads. The generators are connected through step-up transformers to the transmission network at voltage levels of 220 kV and 380 kV. The loads, which represent the distribution networks, are connected at 110 kV buses and via transformers to the transmission network. The parameters of

the network are obtained from the publicly available information from the transmission grid operator[1]. The network model shown in Figure A.2 is developed in the *e*ASiMOV framework and provided as a Matpower case file for extension to dynamic simulation studies in the present thesis.

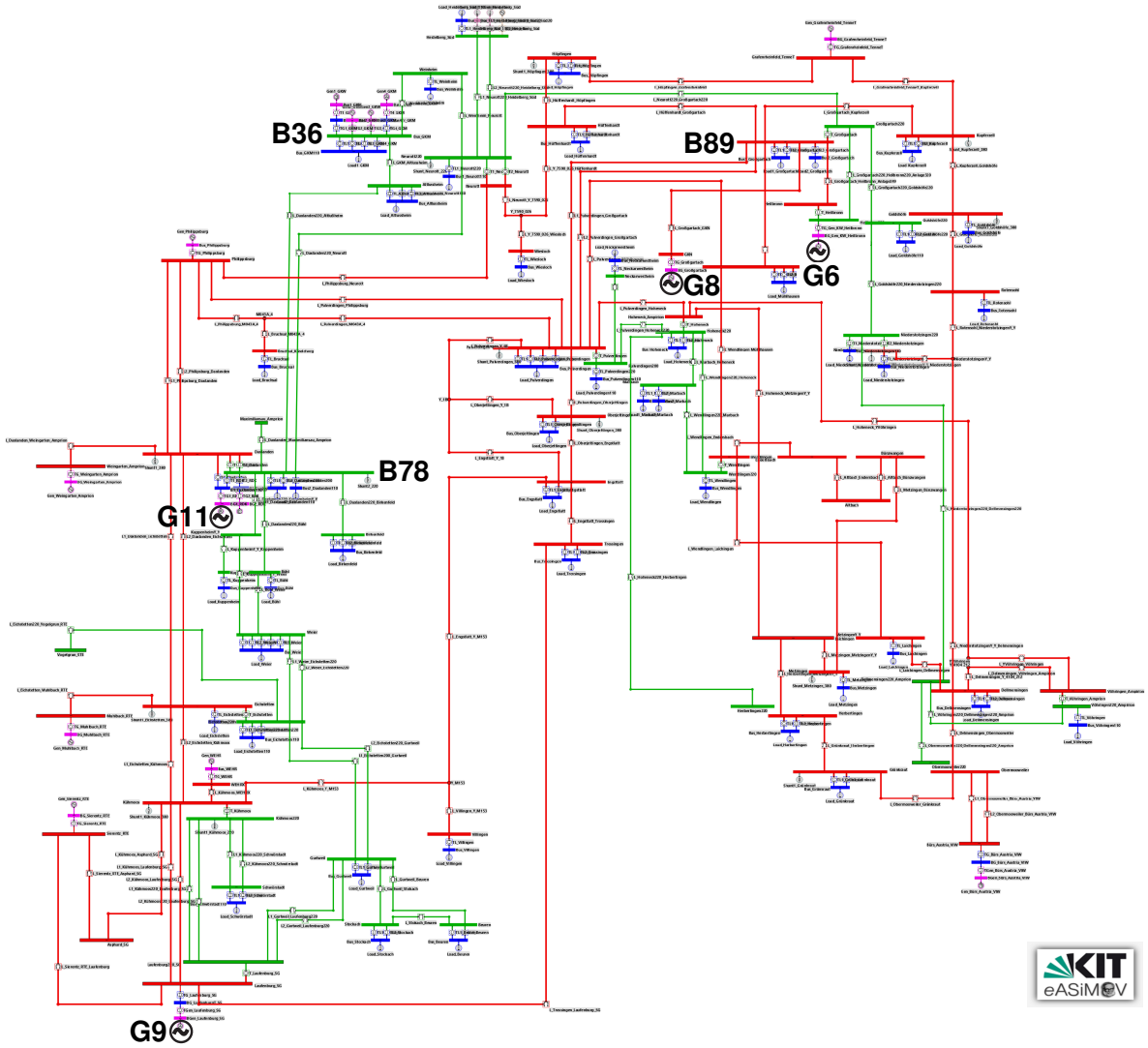


Figure A.2: Structural representation of the Baden-Württemberg transmission network

A.3 Partitioned IEEE 30-Bus Test Network

The test system used for validation of the parallel time-domain computation method is the IEEE 30−bus network. The network consists of 30 buses, 41 branches, 6 generators and 20

[1] https://www.transnetbw.com/en

loads. The steady state network configuration is based on the test network defined in the Matpower packages. The dynamic parameters of the generators are shown in Table A.1, as adapted from [172]. By applying the extended graph partitioning scheme to the network, the resulting optimal partitions for the system divided into three subnetworks are depicted in Figure A.3 (Adapted from[2]), where the buses making up the interconnect partition are 4, 6, 10, 23 and 28.

Table A.1: IEEE 30-Bus network modified dynamic parameters ([172])

Gen Bus	H	D	x_d	x_q	$x_d^{'}$	$x_q^{'}$	$T_d^{'}$	$T_q^{'}$
1	4.130	0.02	1.700	1.620	0.256	0.245	4.800	0.004
2	5.078	0.02	1.270	1.240	0.209	0.850	6.600	0.004
13	1.520	0.02	2.373	1.172	0.343	1.172	11.600	0.159
22	1.520	0.02	2.373	1.172	0.343	1.172	11.600	0.159
23	1.200	0.02	1.769	0.855	0.304	0.5795	8.000	0.008
27	1.200	0.02	1.769	0.855	0.304	0.5795	8.000	0.008

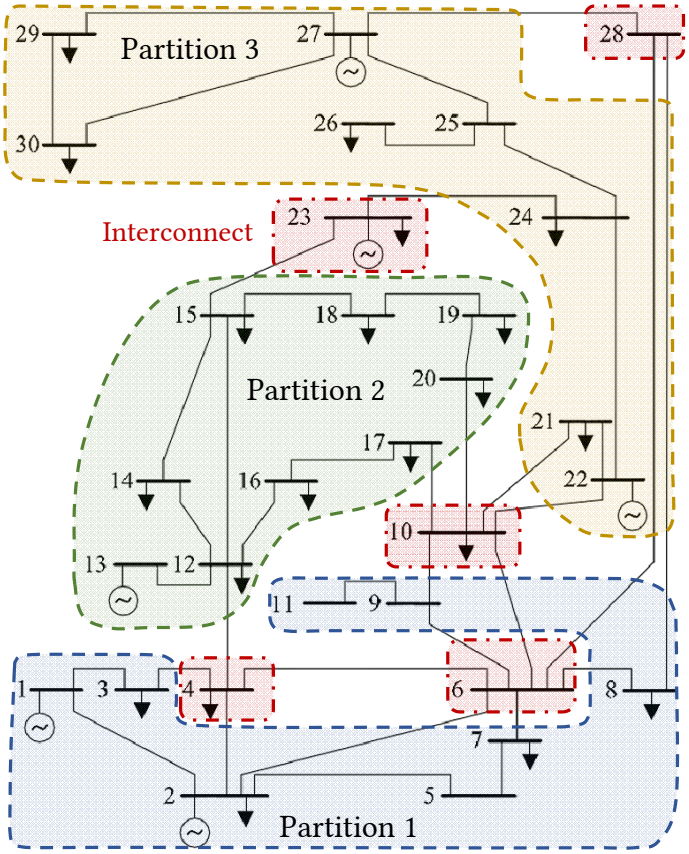


Figure A.3: Partitioned IEEE 30-Bus network with three main partitions and an interconnect partition

[2] https://www.fglongatt.org/Test_Systems/IEEE_30bus.html

B Time-Domain Simulations

B.1 Runtime of Time-Domain Simulation of Conventional Power System

The runtime of the main steps of the time-domain simulation for conventional power systems is summarized in the current section. Table B.1 shows the runtime of the following steps of the simulation: Power flow computation (PF); factorization of the admittance matrix ($FormYbus$); solution of the differential and algebraic equations of the generator subsystem ($Governor$, $Exciter$, $Generator$); calculation for the node current injection ($Isolution$); solution of the network equation for the node voltages ($Vsolution$) and the total computation time ($Runtime$). For further comparison, Figure B.1 depicts the percentage of the runtime taken by the dominant steps of the algorithm for the different networks.

Table B.1: Computational runtime of the main steps of the time-domain simulation algorithm in seconds

Test Network	Powerflow	Form Ybus	Governor	Exciter	Generator	Isolution	Vsolution	Runtime
Case9	0.089	0.003	3.315	1.427	1.480	1.271	0.800	10.785
Case30	0.128	0.003	3.499	1.439	1.492	1.277	0.867	11.115
Case118	0.153	0.004	3.631	1.712	1.744	1.629	1.262	12.603
Case300	0.176	0.010	3.625	1.656	1.671	1.626	2.215	13.455
Case1354	0.160	0.091	4.555	2.557	2.319	3.118	11.128	26.850
Case9241	0.637	9.306	8.458	6.112	5.340	8.800	237.350	280.826
Case13659	0.650	35.714	18.794	14.872	13.165	17.143	498.121	606.719

B.2 Results of Time-domain Simulation with Renewable Energy Sources

Bus Voltage Response to a Network Transient

The present section illustrates the detailed bus voltage response for the simulation of a network fault with and without voltage support for the connected renewable energy generation sources in the Baden-Württemberg network. Initially, the modified network is simulated as described in Section 4.4.1 without voltage support functionality for the connected renewable energy sources during fault. The voltage response at all buses is shown in Figure B.2. The second scenario shown in Figure B.3 illustrates the bus voltage response in the simulation case with renewable energy generators providing fault ride-through and voltage support during the fault period. For illustration purposes, the voltage response of the buses with inverter connected generators are

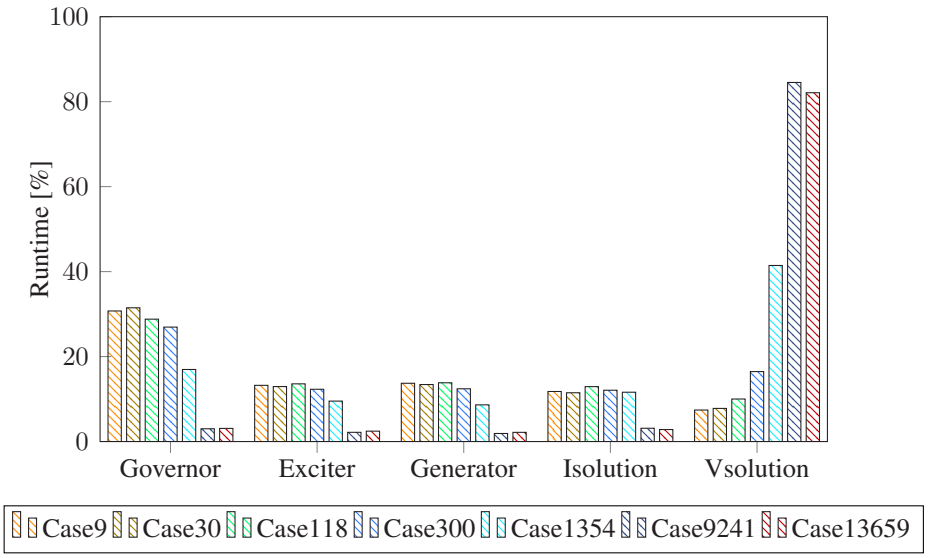


Figure B.1: Percentage of the total runtime for the main stages of the time-domain simulation algorithm

distinguished with dashed lines in Figures B.2 and B.3, while the fault bus voltage is represented using a black dotted line. Important to note is the impact of the voltage support at the inverter connected buses on the other buses depending on the location of the bus relative to the injection point of the reactive power. This impact depends on the location of the buses with respect to the inverter-connected generator providing voltage support.

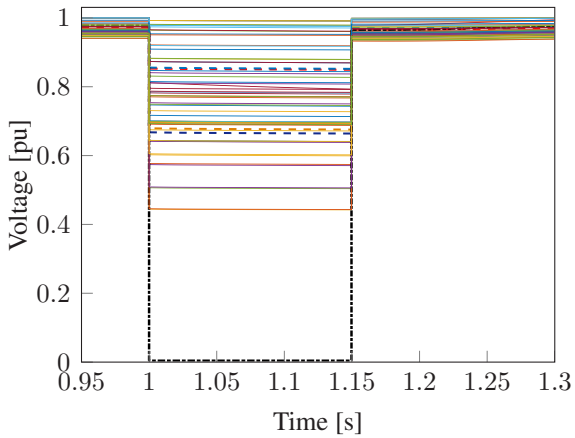


Figure B.2: Bus voltage response for a fault duration of 150 ms without voltage support from PV and wind generation systems: Dashed lines show voltages at the PV and wind generator buses; dotted line represents the fault bus voltage.

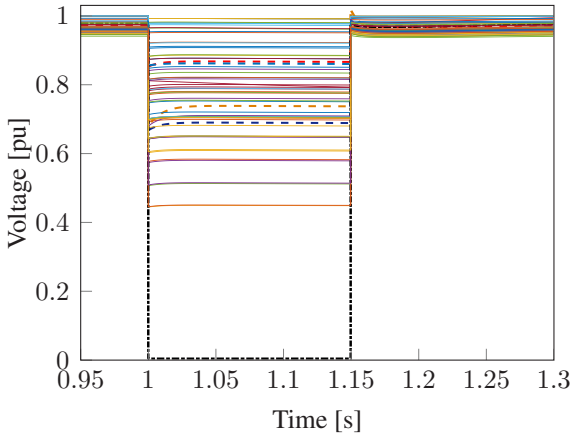


Figure B.3: Bus voltage response with voltage support from PV and wind generation systems during fault: Dashed lines show voltages at the PV and wind generator buses; dotted line represents the fault bus voltage.

B.3 Parallel Time-Domain Simulation

B.3.1 Computation Workflow

The computation process of the parallel time-domain simulation method presented in Chapter 5 is summarized in the workflow depicted in Figure B.4. For illustration purposes, the parallel steps in the simulation process are highlighted in the figure. Part of the precomputation process is the first parallel step, where the admittance matrices are inverted, transposed and corresponding L and U factors are computed in parallel for each partition. Computation of the current injections from the machine differential equations and node voltages from the partitioned network equations at each time step constitute the main parallel steps in the process.

B.3.2 Validation of the Parallel Computation Method

The current section provides further results for the validation presented in Section 5.4.2 comparing the parallel time-domain simulation method to the sequential Matlab-based simulation method. The IEEE 30-bus network partitioned into three subnetworks (cf. Figure A.3) is applied for the presented results. The variables used for the following comparison are generator rotor angle, generator rotor speed, and magnitude of bus voltages as shown in Figures B.5, B.6, and B.7, respectively. For illustration purposes, the voltages shown in Figure B.7 consider only the generator buses and the fault bus. Important to note in all the figures is the perfect match between the sequential and parallel simulation results. This implies that the parallel formulation of the power system problem presented in the current thesis correctly replicates the sequential problem formulation.

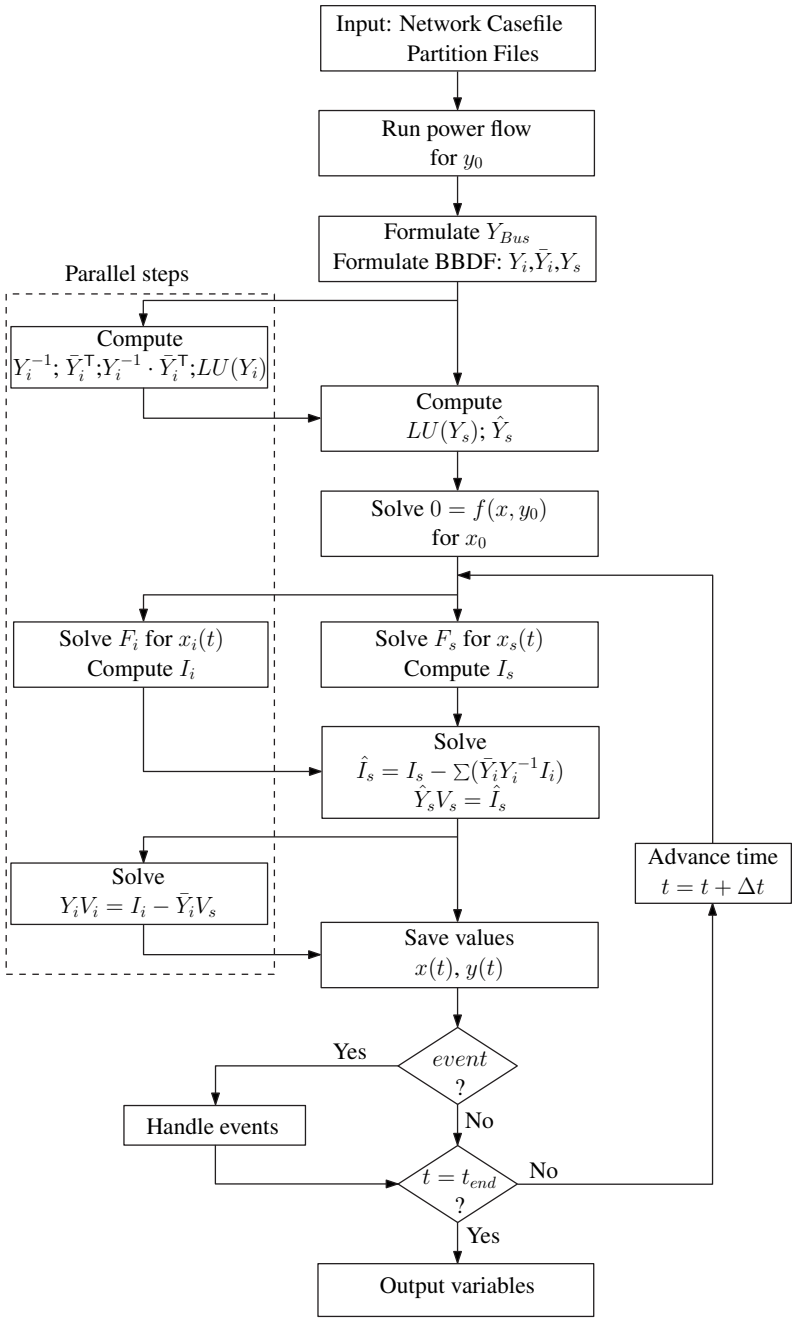


Figure B.4: Parallel time-domain simulation workflow

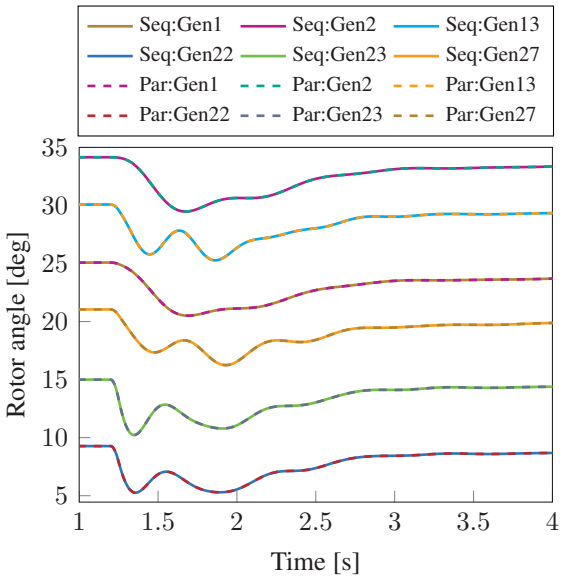


Figure B.5: Comparison of generator rotor angle response to a bus fault in the new parallel time-domain simulation method (Par) and in the sequential Matlab-method (Seq)

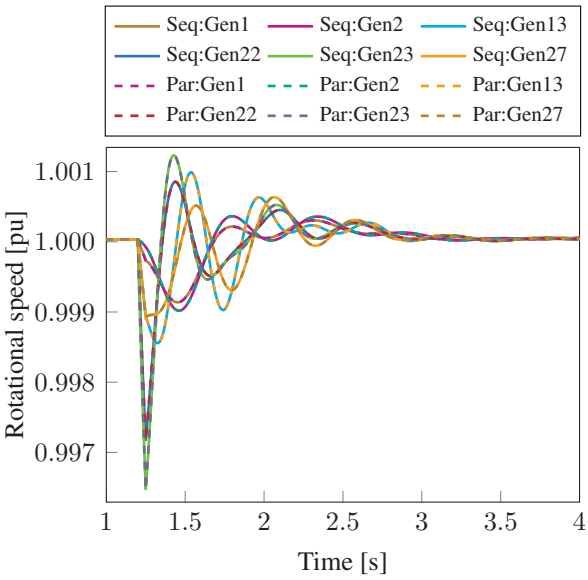


Figure B.6: Comparison of generator rotational speed response to a bus fault in the new parallel time-domain simulation method (Par) and in the sequential Matlab-method (Seq)

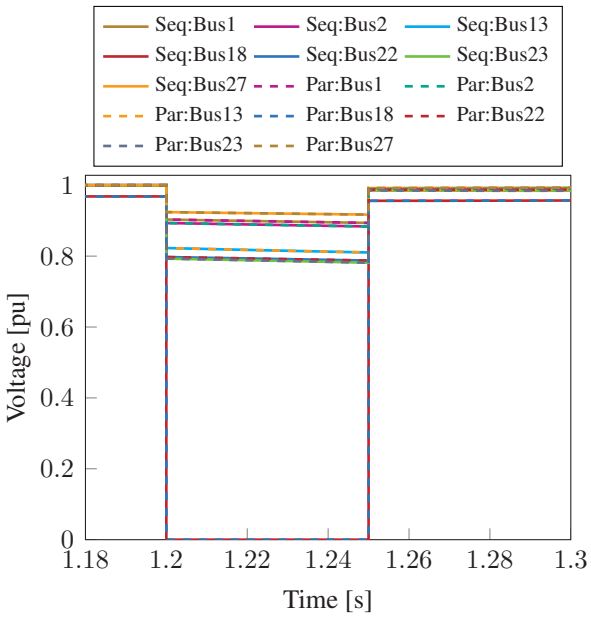


Figure B.7: Comparison of bus voltage response to a bus fault in the new parallel time-domain simulation method (Par) and in the sequential Matlab-method (Seq)

B.3.3 Optimal Partitioning Results and Computational Runtime

In the present section, further partitioning details, runtime, and speedup results of the parallel time-domain simulation method are presented for each test network. Table B.2 summarizes the optimal partitioning for each network and corresponding simulation runtime. The simulation runtime in the Matlab sequential method is also included in Table B.2 for purposes of comparison. Additional runtime and speedup details of tested partitioning counts for each network are shown in Tables B.3 to B.9. Included in the presented results are the average partition and interconnect partition size to illustrate the influence of the optimal network partitioning on the speedup of the parallel computation. In addition, the runtimes of the main stages of the ordinary differential equation (ODE) solver stage, i.e. sequential ODE runtime, parallel ODE runtime, and ODE data input/output time (IO time), are shown to illustrate the main bottleneck in the parallel computation. In summary, the detailed results show that that data OI time is the most time consuming step in the parallel computation process.

Table B.2: Optimal network partitioning count

Network	Partitioning size			Simulation runtime (s)		
	Optimal partitions	Average size	Interconnect size	Matlab Sequential	Julia Sequential	Julia Parallel
Case9	2	3	2	12.81	1.54	1.93
Case30	4	6	6	13.28	1.78	2.11
Case118	6	16	18	15.34	3.95	3.19
Case300	5	57	15	18.52	6.64	4.73
Case1354	7	188	37	48.60	25.22	17.16
Case9241	16	567	161	556.48	212.32	134.84
Case13659	10	1356	97	1274.44	379.63	246.84

Table B.3: Optimal partitioning and runtime evaluation results of Case9

Case9: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	9.0	0	0	0.917	0	0	1.545	0.00
2	3.5	1	2	0.014	0.284	0.983	1.929	-19.94
3	2.0	0	3	0.016	0.295	1.009	1.979	-21.95
4	1.5	1	3	0.018	0.287	1.039	2.080	-25.73

Table B.4: Optimal partitioning and runtime evaluation results of Case30

Case30: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	30.0	0	0	1.079	0	0	1.7843	0.00
2	13.5	1	3	0.0156	0.3834	1.1145	2.2946	-22.24
3	8.7	1	4	0.0184	0.3159	1.2188	2.2965	-22.30
4	6.0	2	6	0.0222	0.3124	1.0856	2.1118	-15.51
5	4.4	1	8	0.0288	0.3185	1.1198	2.1824	-18.24
6	4.0	3	6	0.0265	0.3180	1.2902	2.3958	-25.52
7	3.0	0	9	0.0310	0.3593	1.2163	2.2994	-22.40
8	2.6	1	9	0.0368	0.3556	1.2470	2.3267	-23.31
10	2.1	2	9	0.0413	0.3892	1.3719	2.5026	-28.70

Table B.5: Optimal partitioning and runtime evaluation results of Case118

Case118: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	118.0	0	0	3.009	0	0	3.950	0.00
2	56.0	0	6	0.021	0.694	2.178	3.853	2.51
3	37.3	3	6	0.022	0.636	2.057	4.022	-1.79
4	27.0	2	10	0.031	0.535	1.761	3.319	19.02
5	20.8	1	14	0.030	0.506	1.700	3.189	23.85
6	16.7	1	18	0.051	0.500	1.701	3.188	23.90
7	14.6	3	16	0.055	0.590	2.059	3.662	7.87
8	12.5	2	18	0.060	0.549	1.914	3.487	13.26
10	9.6	2	22	0.074	0.576	2.055	3.676	7.45
12	8.2	3	20	0.078	0.607	2.237	3.878	1.85
14	6.6	2	26	0.102	0.648	2.312	4.035	-2.10
16	5.7	1	27	0.112	0.659	2.393	4.100	-3.67
18	4.9	3	29	0.119	0.687	2.480	4.238	-6.81
20	4.3	3	33	0.143	0.700	2.589	4.378	-9.77

Table B.6: Optimal partitioning and runtime evaluation results of Case300

Case300: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	300.0	0	0	5.356	0	0	6.635	0.00
2	147.5	1	5	0.021	1.213	3.733	6.235	6.41
3	97.0	0	9	0.024	0.984	3.064	5.597	18.54
4	71.8	2	13	0.037	0.882	2.813	4.957	33.85
5	57.0	2	15	0.031	0.831	2.642	4.730	40.28
6	46.5	1	21	0.057	0.843	2.768	4.918	34.91
7	40.1	3	19	0.056	0.895	2.955	5.116	29.70
8	34.6	3	23	0.073	0.884	2.974	5.198	27.65
10	27.6	2	24	0.082	0.913	3.001	5.242	26.57
12	22.2	3	34	0.116	0.957	3.260	5.533	19.91
14	19.1	4	32	0.122	0.960	3.306	5.733	15.73
16	16.6	3	35	0.138	1.003	3.461	5.857	13.28
18	14.3	5	43	0.174	1.005	3.674	6.072	9.27
20	12.4	3	53	0.226	0.981	3.809	6.226	6.57

Table B.7: Optimal partitioning and runtime evaluation results of Case1354

Case1354: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	1354.0	0	0	21.755	0	0	25.219	0.00
2	671.0	0	12	0.029	5.499	18.200	27.824	-9.36
3	444.3	1	21	0.051	4.580	14.792	23.476	7.42
4	333.5	19	20	0.052	3.654	12.570	19.475	29.49
5	264.2	11	33	0.084	3.465	10.676	18.966	32.97
6	218.8	10	41	0.110	3.289	11.903	18.557	35.90
7	188.1	5	37	0.110	3.388	10.419	17.161	46.96
8	164.4	21	39	0.120	3.029	11.229	18.039	39.81
10	130.0	18	54	0.181	3.351	14.139	20.879	20.79
12	107.4	10	65	0.230	3.225	10.020	20.060	25.72
14	91.1	4	78	0.281	3.156	10.287	19.097	32.06
16	80.0	7	74	0.300	3.260	10.790	17.810	41.60
18	70.2	6	90	0.389	3.262	11.332	20.380	23.74
20	63.0	4	94	0.412	3.203	10.939	18.117	39.20

Table B.8: Optimal partitioning and runtime evaluation results of Case9241

Case9241: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	9241.0	0	0	180.663	0	0	212.317	0.00
2	4613.0	222	15	0.060	43.919	136.900	234.719	-9.54
3	3070.7	251	29	0.204	40.263	120.301	204.363	3.89
4	2297.8	66	50	0.296	39.087	114.632	191.643	10.79
5	1837.2	118	55	0.316	34.871	109.435	176.076	20.58
6	1526.8	65	80	0.395	33.122	108.485	174.022	22.01
7	1309.7	78	73	0.397	31.181	99.428	161.857	31.18
8	1145.5	55	77	0.416	29.204	88.995	148.521	42.95
10	913.9	37	102	0.544	24.434	82.673	141.081	50.49
12	760.1	34	120	0.631	28.016	73.709	137.464	54.45
14	649.9	54	143	0.773	23.559	77.790	136.700	55.32
16	567.5	41	161	0.865	22.690	72.280	134.839	57.46
18	503.3	16	181	1.047	26.310	70.334	134.871	57.42
20	451.2	39	217	1.474	22.731	83.427	140.697	50.90

Table B.9: Optimal partitioning and runtime evaluation results of Case13659

Case13659: Partition size				ODE stage runtime [s]			Simulation	
Partition count	Average size	Partition difference	Interconnect size	Sequential ODE	Parallel ODE	ODE DataIO	Runtime [s]	Speedup %
1	13659.0	0	0	323.71	0	0	379.632	0.00
2	6822.5	29	14	0.098	76.798	227.211	421.405	-9.91
3	4542.7	306	31	0.278	66.659	205.913	370.195	2.55
4	3403.3	93	46	0.348	58.752	189.362	326.571	16.25
5	2722.6	173	46	0.367	51.302	166.255	289.894	30.96
6	2264.5	155	72	0.451	56.668	170.814	298.786	27.06
7	1941.1	76	71	0.469	50.373	154.533	271.904	39.62
8	1696.0	152	91	0.550	49.378	155.393	277.467	36.82
10	1356.2	85	97	0.614	49.800	131.221	246.839	53.80
12	1127.5	174	129	0.800	43.864	140.100	250.037	51.83
14	964.1	76	162	1.034	41.157	143.957	264.546	43.50
16	843.3	59	167	1.154	43.452	135.616	254.699	49.05
18	748.7	65	182	1.386	38.522	128.556	249.881	51.92
20	673.0	84	200	1.674	38.326	131.923	249.951	51.88

C Hybrid Method and System Integration

C.1 Direct Method Runtime

Table C.1 summarizes the performance of the main steps of the direct method presented in Chapter 6 in terms of the total runtime for all contingencies as described in Table 6.1. The fault duration in each case is 200 ms. It should be noted that the runtime in this case includes some overhead resulting from profiling of the analysis process.

Table C.1: Total runtime of the main steps of the proposed method with varying network complexity

Test Network	Ybus Formulation	Network Reduction	Eigenvalue Analysis	Time-domain Simulation	Contingency Selection	Total Runtime
Case68	0.031	0.767	0.050	1.861	0.220	9.713
CaseBW	0.104	1.470	0.150	5.380	0.612	24.725
Case118	0.197	1.332	1.344	6.349	1.108	26.541
Case1354	139.406	289.752	473.407	321.983	148.184	2144.050
CaseDE	700.443	647.561	2339.516	1499.890	1186.919	7684.786

C.2 Analysis in the Integrated Software Framework

Overview of the *e*ASiMOV Framework

The *e*ASiMOV framework was developed as part of the research and development work at the Institute for Automation and Applied Informatics at Karlsruhe Institute of Technology. The main components of the *e*ASiMOV framework are summarized in Figure C.1. *e*PowMod is the modeling component of the framework. The module provides a Java-based interactive grid modeling editor with support for schematic and geo-based modeling. *e*PowSim is the simulation software module with CPU, GPU, multicore and cluster simulation capabilities. The *e*PowVis module provides the analysis and visualization function for the power grid simulation results. The additional modules in the framework include; *e*PowMgr – a Java EE-Server application for managing network storages, models and time series data, user database and external simulators. The module additionally supports sharing of models, time series data and simulation results via public hyperlinks; *e*PowWeb – the web based simulation and visualization module; *e*PowCon – the data conversion toolbox which enables exchange of models with external software packages.

The current state of the *e*PowCon provides direct import and export functionalities for Excel based data sheets, and model data for OpenDSS, Matpower and DIgSILENT PowerFactory. Other add-on software modules to the framework providing services such as analysis and interactive exploration of high rate energy data, web based energy data monitoring, web services for

load data aggregation, remote software control as well as support for mobile devices are also part of the software suite.

As described in Section 7.2, the methods developed in the present thesis contribute to the simulation module of the *e*ASiMOV framework (*e*PowSim). This therefore extends the simulation capability of the framework to dynamic simulations.

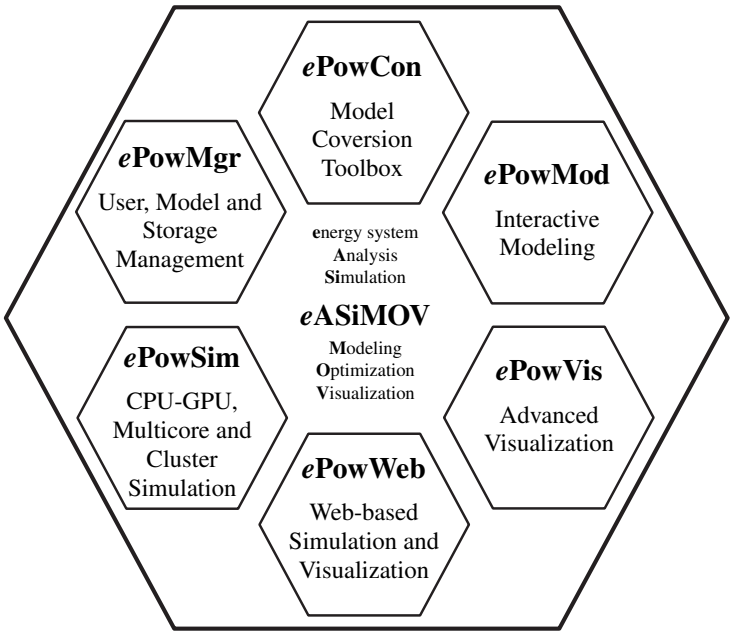
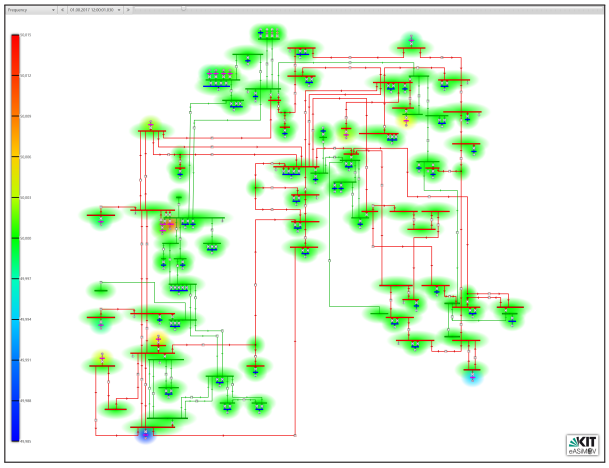


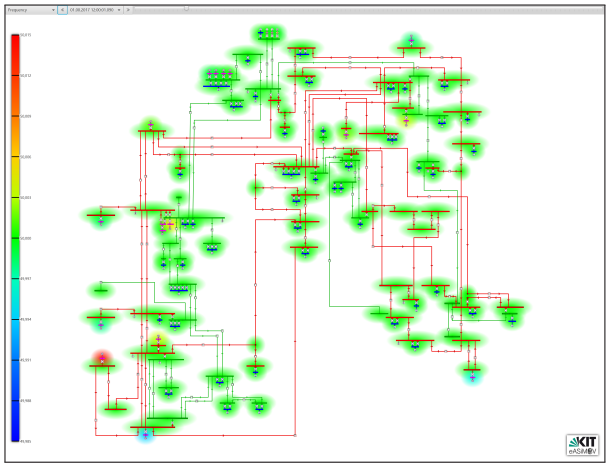
Figure C.1: Overview of the modules in the *e*ASiMOV framework

Visualization in the *e*ASiMOV Framework

The current section presents the detailed visualization results of the Baden-Württemberg network in the *e*ASiMOV framework as part of the integrated software framework. Figures C.2 to C.4 show the detailed visualization of the system frequency in the respective cases analyzed in Section 7.3 for the unclassifiable and critical contingencies.

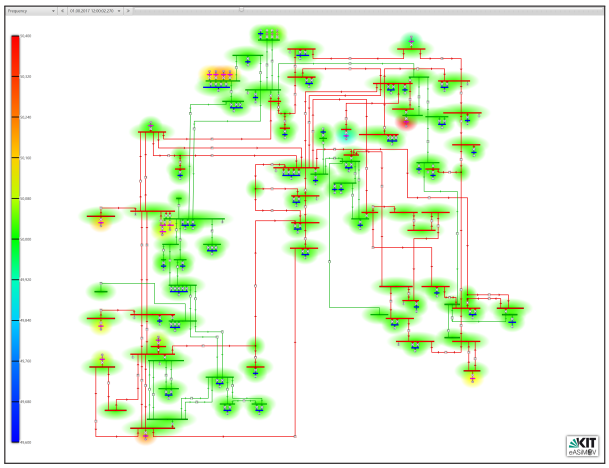


(a) Visualization of system frequency at $t = 1.030\,\mathrm{s}$

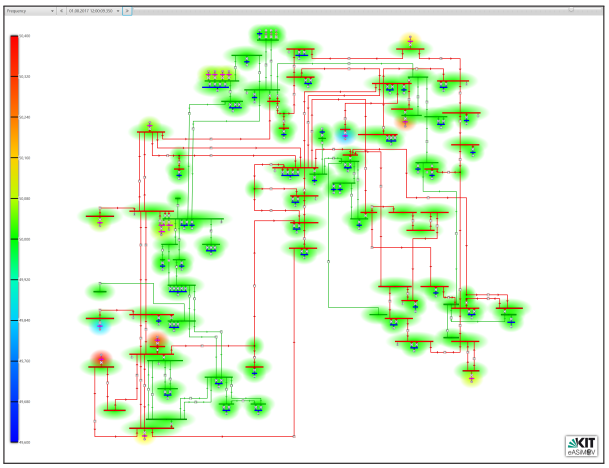


(b) Visualization of system frequency at $t = 1.090\,\mathrm{s}$

Figure C.2: Interactive visualization of system frequency in the *e*ASiMOV framework for a very critical switching case. Frequency scale ranges from $49.985\,\mathrm{Hz}$ to $50.015\,\mathrm{Hz}$.

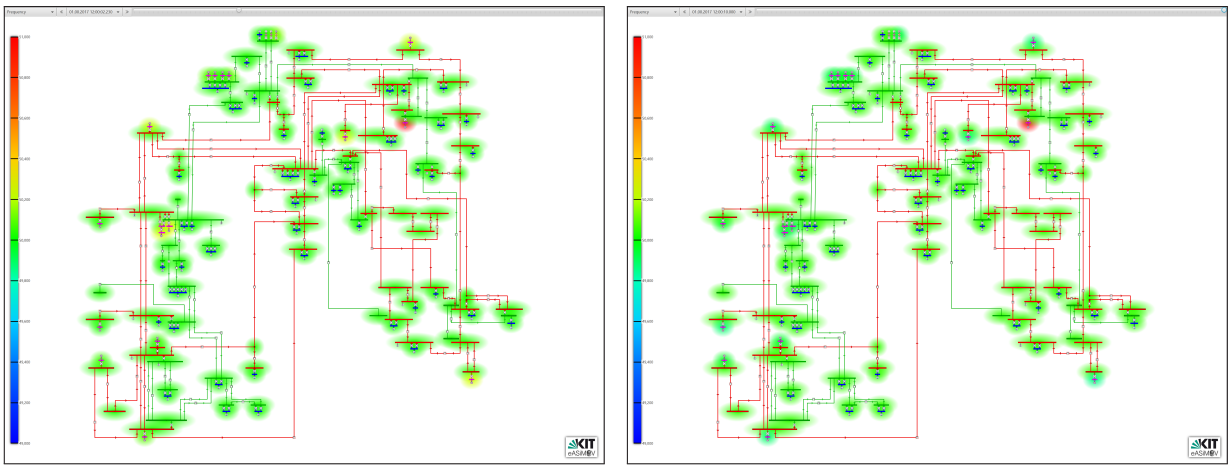


(a) Visualization of system frequency at $t = 2.270\,\mathrm{s}$



(b) Visualization of system frequency at $t = 9.350\,\mathrm{s}$

Figure C.3: Interactive visualization of system frequency in the *e*ASiMOV framework for a fault cleared before the critical clearing point. Frequency scale ranges from $49.6\,\mathrm{Hz}$ to $50.4\,\mathrm{Hz}$.



(a) Visualization of system frequency at $t = 2.230\,\mathrm{s}$ (b) Visualization of system frequency at $t = 10.00\,\mathrm{s}$

Figure C.4: Interactive visualization of system frequency in the *e*ASiMOV framework for a fault cleared at a time beyond critical clearing time. Frequency scale ranges from $49.0\,\mathrm{Hz}$ to $51.0\,\mathrm{Hz}$.

List of Figures

List of Tables

Bibliography

[1] UNEP (2017), "The Emissions Gap Report 2017," A UN Environment Synthesis Report, United Nations Environment Programme (UNEP), November 2017.

[2] I. Graabak and M. Korpås, "Variability Characteristics of European Wind and Solar Power Resources–A Review," *Energies*, vol. 9, no. 6, 2016.

[3] IRENA (2018), "Power System Flexibility for the Energy Transition," Part 1: Overview for Policy Makers, International Renewable Energy Agency, November 2018.

[4] Fraunhofer ISE, "Energy Charts." https://www.energy-charts.de/power_inst.htm. Accessed: 25 April 2019.

[5] IRENA (2019), "Renewable Capacity Statistics 2019," International Renewable Energy Agency, March 2019.

[6] REN21 (2018), "Renewables 2018 Global Status Report," Paris: REN21 Secretariat, 2018.

[7] T. Bründlinger, J. Elizalde-König, O. Frank, and et al., "The dena Study – Integrated Energy Transition," Deutsche Energie-Agentur GmbH (dena), Berlin, October 2018.

[8] M. Huber, D. Dimkova, and T. Hamacher, "Integration of Wind and Solar Power in Europe: Assessment of Flexibility Requirements," *Energy*, vol. 69, pp. 236–246, 2014.

[9] S. D'Arco, J. A. Suul, and O. B. Fosso, "A Virtual Synchronous Machine Implementation for Distributed Control of Power Converters in SmartGrids," *Electric Power Systems Research*, vol. 122, pp. 180–197, 2015.

[10] G. Lammert, K. Yamashita, L. D. P. Ospina, H. Renner, S. M. Villanueva, P. Pourbeik, F.-E. Ciausiu, and M. Braun, "International Industry Practice on Modelling and Dynamic Performance of Inverter Based Generation in Power System Studies," *CIGRE Science and Engineering*, vol. 8, pp. 25–37, 2017.

[11] P. Palensky and D. Dietrich, "Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 3, pp. 381–88, 2011.

[12] P. Siano, "Demand Response and Smart Grids–A Survey," *Renewable and Sustainable Energy Review*, vol. 30, pp. 461–478, 2014.

[13] I. Colak, G. Fulli, S. Sagiroglu, M. Yesilbudak, and C.-F. Covrig, "Smart Grid Projects in Europe: Current Status, Maturity and Future Scenarios," *Applied Energy*, vol. 152, pp. 58–70, 2015.

[14] V. Hagenmeyer, H. K. Çakmak, C. Düpmeier, T. Faulwasser, J. Isele, H. B. Keller, P. Kohlhepp, U. Kühnapfel, U. Stucky, S. Waczowicz, and R. Mikut, "Information and Communication Technology in Energy Lab 2.0: Smart Energies System Simulation and Control Center with an Open-Street-Map-Based Power Flow Simulation Example," *Energy Technology*, vol. 4, no. 1, pp. 145–162, 2016.

[15] J. Blanger, P. Venne, and J.-N. Paquin, "The What, Where and Why of Real-Time Simulation," in *IEEE PES General Meeting*, Minneapolis MN, USA, July 2010.

[16] D. Povh, D. Retzmann, and J. Rittiger, "Benefits of Simulation for Operation of Large Power Systems and System Interconnections," in *Proceedings of the 4^{th} IERE General Meeting*, Krakow, Poland, October 2004.

[17] F. Milano, *Power System Modelling and Scripting*. Heidelberg: Springer, 2010.

[18] P. M. Anderson and A. A. Fouad, *Power System Control and Stability*. Piscataway, NJ: IEEE Press, 2003.

[19] IEEE/CIGRE joint task force on stability terms and definitions, "Definition and Classification of Power System Stability," *IEEE Transactions on Power Systems*, vol. 19, no. 3, pp. 1387–1401, 2004.

[20] IEEE Task Force on Interfacing Techniques for Simulation Tools, "Interfacing Techniques for Transient Stability and Electromagnetic Transient Programs ," *IEEE Transactions on Power Delivery*, vol. 24, no. 4, pp. 2385–2395, 2009.

[21] J. Mahseredjian, V. Dinavahi, and J. A. Martinez, "Simulation Tools for Electromagnetic Transients in Power Systems: Overview and Challenges," *IEEE Transactions on Power Delivery*, vol. 24, no. 3, pp. 1657–1669, 2009.

[22] C4.601 CIGRE Working group, "Review of On-Line Dynamic Security Assessment Tools and Techniques," Technical Brochure 325, CIGRE, Paris, June 2007.

[23] J. Machowski, J. W. Bialek, and J. R. Bumby, *Power System Dynamics : Stability and Control*. Chichester: Wiley, 2008.

[24] H. W. Dommel, *Electromagnetic Transient Program (EMTP) Theory Book*. Portland, Oregon: Bonneville Power Administration, 1986.

[25] Manitoba HVDC Research Centre, *User's Guide EMTDC^{TM} Transient Analysis for PSCAD Power System Simulation*, 2016.

[26] RTDS Technologies, *Real Time Digital Simulation for the Power Industry*. Winnipeg, Canada, 2018.

[27] Hydro-Québec and T. Technologies, *SimPowerSystems, for use with Simulink*. TransÉnergie Technologies Inc, 2003.

[28] Siemens Power Transmission Distribution, Inc, *PSS/E User Manual*, December 2007.

[29] DIgSILENT GmbH, *DIgSILENT PowerFactory User Manual Version 2016*. Gomaringen, Germany, March 2016.

[30] BCP, *Neplan-Power System Analysis Software*. http://www.neplan.ch, 2007.

[31] F. Milano and L. Vanfretti, "State of the Art and Future of OSS for Power Systems," in *2009 IEEE Power Energy Society General Meeting*, Calgary, AB, Canada, July 2009.

[32] D. P. Chassin, K. Schneider, and C. Gerkensmeyer, "GridLAB-D: An Open-Source Power Systems Modeling and Simulation Environment," in *2008 IEEE/PES Transmission and Distribution Conference and Exposition*, 2008.

[33] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, 2011.

[34] R. Lincoln, "PYPOWER." https://pypi.org/project/PYPOWER/4.0.0/l.

[35] T. Brown, J. Hörsch, and D. Schlachtberger, "PyPSA: Python for Power System Analysis," *Journal of Open Research Software*, vol. 6, no. 1, 2018.

[36] F. Milano, "An Open Source Power System Analysis Toolbox," *IEEE Transactions on Power Systems*, vol. 20, no. 3, pp. 1199–1206, 2005.

[37] M. Zhou, "Distributed Parallel Power System Simulation," in *High Performance Computing in Power and Energy Systems*, pp. 71–100, Springer Berlin Heidelberg, 2013.

[38] P. Fritzson, P. Aronsson, H. Lundvall, K. Nyström, A. Pop, L. Saldamli, and D. Broman, "The OpenModelica Modeling, Simulation, and Software Development Environment," *Simulation News Europe*, vol. 44, pp. 8–16, December 2005.

[39] R. C. Dugan and T. E. McDermott, "An Open Source Platform for Collaborating on Smart Grid Research," in *2011 IEEE Power and Energy Society General Meeting*, July 2011.

[40] S. Cole and R. Belmans, "MatDyn, A New Matlab-Based Toolbox for Power System Dynamic Simulation," *IEEE Transactions on Power Systems*, vol. 26, pp. 1129–1136, August 2011.

[41] S. Vera, "GridCal-Research Oriented Power Systems Software." https://github.com/SanPen/GridCal.

[42] F. Milano, "A Python-Based Software Tool for Power System Analysis," in *2013 IEEE Power Energy Society General Meeting*, July 2013.

[43] J. Susanto, "pypower-dynamics: Time-Domain Simulation (Transient Stability) Module for PYPOWER." https://pypi.org/project/pypower-dynamics/.

[44] L. Thurner, A. Scheidler, F. Schäfer, J. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun, "Pandapower–An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6510–6521, 2018.

[45] T. H. Chen, M. S. Chen, T. Inoue, P. Kotas, and E. A. Chebli, "Three-Phase Cogenerator and Transformer Models for Distribution System Analysis," *IEEE Transactions on Power Delivery*, vol. 6, no. 4, pp. 1671–1681, 1991.

[46] S. M. Halpin, C. A. Gross, and L. L. Grigsby, "An Improved Method of Including Detailed Synchronous Machine Representations in Large Power System Models for Fault Analysis," *IEEE Transactions on Energy Conversion*, vol. 8, no. 4, pp. 719–725, 1993.

[47] J. Tamura, I. Takeda, M. Kimura, and S. Yonaga, "A Synchronous Machine Model for Unbalanced Analyses," *Electrical Engineering in Japan*, vol. 119, no. 2, pp. 46–59, 1997.

[48] M. Karimi-Ghartemani and H. Karimi, "Analysis of Symmetrical Components in Time-Domain," in *48^{th} Midwest Symposium on Circuits and Systems*, August 2005.

[49] R. Kumar, B. Singh, and D. T. Shahani, "Symmetrical Components-Based Modified Technique for Power-Quality Disturbances Detection and Classification," *IEEE Transactions on Industry Applications*, vol. 52, no. 4, pp. 3443–3450, 2016.

[50] M. Abdel-Akher, K. M. Nor, and A. H. A. Rashid, "Improved Three-Phase Power-Flow Methods using Sequence Components," *IEEE Transactions on Power Systems*, vol. 20, no. 3, pp. 1389–1397, 2005.

[51] M. Z. Kamh and R. Iravani, "Unbalanced Model and Power-Flow Analysis of Microgrids and Active Distribution Systems," *IEEE Transactions on Power Delivery*, vol. 25, no. 4, pp. 2851–2858, 2010.

[52] E. Demirok, S. B. Kjær, D. Sera, and R. Teodorescu, "Three-Phase Unbalanced Load Flow Tool for Distribution Networks," in *2^{nd} International Workshop on Integration of Solar Power Systems*, 2012.

[53] R. H. Salim and R. A. Ramos, "A Model-Based Approach for Small-Signal Stability Assessment of Unbalanced Power Systems," *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 2006–2014, 2012.

[54] S. Saha and M. Aldeen, "Dynamic Modeling of Power Systems Experiencing Faults in Transmission/Distribution Networks," *IEEE Transactions on Power Systems*, vol. 30, no. 5, pp. 2349–2363, 2015.

[55] M. A. Elizondo, F. K. Tuffner, and K. P. Schneider, "Three-Phase Unbalanced Transient Dynamics and Powerflow for Modeling Distribution Systems With Synchronous Machines," *IEEE Transactions on Power Systems*, vol. 31, no. 1, pp. 105–115, 2016.

[56] A. M. Stankovic and T. Aydin, "Analysis of Asymmetrical Faults in Power Systems using Dynamic Phasors," *IEEE Transactions on Power Systems*, vol. 15, no. 3, pp. 1062–1068, 2000.

[57] A. M. Stankovic, S. R. Sanders, and T. Aydin, "Dynamic Phasors in Modeling and Analysis of Unbalanced Polyphase AC Machines," *IEEE Transactions on Energy Conversion*, vol. 17, no. 1, pp. 107–113, 2002.

[58] T. Demiray, G. Andersson, and L. Busarello, "Evaluation Study for the Simulation of Power System Transients using Dynamic Phasor Models," in *IEEE/PES Transmission and Distribution Conference and Exposition: Latin America*, Bogota, Colombia 2008.

[59] S. Abhyankar and A. J. Flueck, "An Implicitly-Coupled Solution Approach for Combined Electromechanical and Electromagnetic Transients Simulation," in *2012 IEEE Power and Energy Society General Meeting*, San Diego, CA, July 2012.

[60] F. J. Plumier, P. Aristidou, C. Geuzaine, and T. V. Cutsem, "A Relaxation Scheme to Combine Phasor-Mode and Electromagnetic Transients Simulations," in *2014 Power Systems Computation Conference*, Wroclaw, Poland, August 2014.

[61] M. Sultan, J. Reeve, and R. Adapa, "Combined Transient and Dynamic Analysis of HVDC and FACTS Systems," *IEEE Transactions on Power Delivery*, vol. 13, no. 4, pp. 1271–1277, 1998.

[62] X. Wang, P. Wilson, and D. Woodford, "Interfacing Transient Stability Program to EMTDC Program," in *International Confeference on Power System Technology*, Kunming, China, October 2002.

[63] Y. Zhang, A. M. Gole, W. Wu, B. Zhang, and H. Sun, "Development and Analysis of Applicability of a Hybrid Transient Simulation Platform Combining TSA and EMT Elements," *IEEE Transaction on Power Systems*, vol. 28, no. 1, pp. 357–366, 2013.

[64] A. A. van der Meer, M. Gibescu, M. A. M. M. van der Meijden, W. L. Kling, and J. A. Ferreira, "Advanced Hybrid Transient Stability and EMT Simulation for VSC-HVDC Systems," *IEEE Transactions on Power Delivery*, vol. 30, no. 3, pp. 1057–1066, 2015.

[65] S. Oh and S. Chae, "A Co-Simulation Framework for Power System Analysis," *Energies*, vol. 9, no. 3, p. 131, 2016.

[66] P. Aristidou and T. V. Cutsem, "Dynamic Simulations of Combined Transmission and Distribution Systems using Parallel Processing Technique," in *Power Systems Computation Conference*, Wroclaw, Poland, August 2014.

[67] P. Aristidou and T. V. Cutsem, "A Parallel Processing Approach to Dynamic Simulations of Combined Transmission and Distribution Systems," *International Journal of Electrical Power & Energy Systems*, vol. 72, pp. 58–65, 2015.

[68] H. Jain, A. Parchure, R. P. Broadwater, M. Dilek, and J. Woyak, "Three-Phase Dynamic Simulation of Power Systems Using Combined Transmission and Distribution System Models," *IEEE Transactions on Power Systems*, vol. 31, no. 6, pp. 4517–4524, 2016.

[69] R. Huang, R. Fan, J. Daily, A. Fisher, and J. Fuller, "An Open-source Framework for Power System Transmission and Distribution Dynamics Co-simulation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, no. 12, pp. 3152–3162, 2017.

[70] D. P. Koester, S. Ranka, and G. C. Fox, "Power Systems Transient Stability – A Grand Computing Challenge," in *Proceedings of the Northeast Parallel Architectures Center (NPAC)*, Syracuse, New York, August 1992.

[71] M. La Scala, G. Sblendorio, A. Bose, and J. Q. Wu, "Comparison of Algorithms for Transient Stability Simulations on Shared and Distributed Memory Multiprocessors," *IEEE Transactions on Power Systems*, vol. 11, no. 4, pp. 2045–2050, 1996.

[72] C. Dufour, V. Jalili-Marandi, J. Bélanger, and L. Snider, "Power System Simulation Algorithms for Parallel Computer Architectures," in *2012 IEEE Power and Energy Society General Meeting*, July 2012.

[73] D. M. Falcao, "High Performance Computing in Power System Applications," in *International Conference on Vector and Parallel Processing*, Porto, Portugal 1997.

[74] R. C. Green, L. Wang, and M. Alam, "High Performance Computing for Electric Power Systems: Applications and Trends," in *2011 IEEE Power and Energy Society General Meeting*, San Diego, CA, July 2011.

[75] R. C. Green, L. Wang, and M. Alam, "Applications and Trends of High Performance Computing for Electric Power Systems: Focusing on Smart Grid," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 922–931, 2013.

[76] I. Decker, D. Falcao, and E. Kaszkurewicz, "An Efficient Parallel Method for Transient Stability Analysis," in *Tenth Power Systems Computation Conference*, 1990.

[77] I. Decker, D. Falcao, and E. Kaszkurewicz, "Parallel Implementation of a Power System Dynamic Simulation Methodology using the Conjugate Gradient Method," in *Power Industry Computer Application Conference*, Baltimore, MD, US, May 1991.

[78] I. Decker, D. Falcao, and E. Kaszkurewicz, "Conjugate Gradient Methods for Power System Dynamic Simulation on Parallel Computers," *IEEE Transactions on Power Systems*, vol. 11, no. 3, pp. 1218–1227, 1996.

[79] S.Esmaeili and S.M.Kouhsari, "A Distributed Simulation Based Approach for Detailed and Decentralized Power System Transient Stability Analysis," *Electric Power Systems Research*, vol. 77, no. 5-6, pp. 673–684, 2007.

[80] M. Tomim, J. Martí, and L. Wang, "Parallel Solution of Large Power System Networks using the Multi-Area Thévenin Equivalents (MATE) Algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 31, no. 9, pp. 497–503, 2009.

[81] S. Abhyankar, A. Flueck, X. Zhang, and H. Zhang, "Development of a Parallel Three-Phase Transient Stability Simulator for Power Systems," in *Proceedings of the first international workshop on High performance computing, networking and analytics for the power grid*, Seattle, Washington, USA, November 2011.

[82] S. Abhyankar and A. Flueck, "Real-Time Power System Dynamics Simulation using a Parallel Block-Jacobi Preconditioned Newton-GMRES Scheme," in *2012 SC Companion: High Performance Computing, Networking Storage and Analysis*, Salt Lake City, UT, November 2012.

[83] M. La Scala, R. Sbrizzai, and F. Torelli, "A Pipelined-in-Time Parallel Algorithm for Transient Stability Analysis (Power Systems)," *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 715–722, 1991.

[84] F. Z. Wang, "Parallel-in-Time Relaxed Newton Method for Transient Stability Analysis," *IEE Proceedings – Generation, Transmission and Distribution*, vol. 145, no. 2, pp. 155–159, 1998.

[85] M. La Scala, R. Sbrizzai, F. Torelli, and P. Scarpellini, "A Tracking Time Domain Simulator for Real-Time Transient Stability Analysis," *IEEE Transactions on Power Systems*, vol. 13, no. 3, pp. 992–998, 1998.

[86] M. Ilic'-Spong, M. L. Crow, and M. A. Pai, "Transient Stability Simulation by Waveform Relaxation Methods," *IEEE Transactions on Power Systems*, vol. 2, no. 4, pp. 943–949, 1987.

[87] M. L. Crow and M. Ilic, "The Parallel Implementation of the Waveform Relaxation Method for Transient Stability Simulations," *IEEE Transactions on Power Systems*, vol. 5, no. 3, pp. 922–932, 1990.

[88] M. La Scala, A. Bose, D. J. Tylavsky, and J. S. Chai, "A Highly Parallel Method for Transient Stability Analysis," *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1439–1446, 1990.

[89] L. Hou and A. Bose, "Implementation of the Waveform Relaxation Algorithm on a Shared Memory Computer for the Transient Stability Problem," *IEEE Transactions on Power Systems*, vol. 12, no. 3, pp. 1053–1060, 1997.

[90] V. Jalili-Marandi and V. Dinavahi, "Instantaneous Relaxation-Based Real-Time Transient Stability Simulation," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1327–1336, 2009.

[91] S. Abhyankar and A. Flueck, "Parallel-in-Space-and-Time Scheme for Implicitly Coupled Electromechanical and Electromagnetic Transients Simulation," in *International Conference on Power Systems*, Vancouver, Canada, July 2013.

[92] S. K. Khaitan and A. Gupta, "High Performance Computing for Power System Dynamic Simulation," in *High Performance Computing in Power and Energy Systems*, pp. 43–69, Springer Berlin Heidelberg, 2013.

[93] H.-D. Chiang, C.-C. Chu, and G. Cauley, "Direct Stability Analysis of Electric Power Systems using Energy Functions: Theory, Applications, and Perspective," *Proceedings of the IEEE*, vol. 83, no. 11, pp. 1497–1529, 1995.

[94] L. F. C. Alberto, F. H. J. R. Silva, and N. G. Bretas, "Direct Methods for Transient Stability Analysis in Power Systems: State of Art and Future Perspectives," in *2001 IEEE Porto Power Tech Proceedings*, vol. 2, September 2001.

[95] M. Eremia and M. Shahidehpour, *Handbook of Electrical Power System Dynamics : Modeling, Stability, and Control*. Hoboken, NJ: IEEE Wiley, 2013.

[96] P. Kundur, *Power System Stability and Control*. New York: McGraw-Hill, Inc, 1994.

[97] H. H. Al Marhoon, I. Leevongwat, and P. Rastgoufard, "A Practical Method for Power Systems Transient Stability and Security Analysis," in *PES T&D 2012*, Orlando, FL, USA, May 2012.

[98] Y. Zhang, L. Wehenkel, P. Rousseaux, and M. Pavella, "SIME: A Hybrid Approach to Fast Transient Stability Assessment and Contingency Selection," *International Journal of Electrical Power & Energy Systems*, vol. 19, no. 3, pp. 195–208, 1997.

[99] Y. Zhang, L. Wehenkel, and M. Pavella, "SIME: A Comprehensive Approach to Fast Transient Stability Assessment," *IEEJ Transactions on Power and Energy*, vol. 118, no. 2, pp. 127–132, 1998.

[100] D. Ernst, D. Ruiz-Vega, M. Pavella, P. M. Hirsch, and D. Sobajic, "A Unified Approach to Transient Stability Contingency Filtering, Ranking and Assessment," *IEEE Transactions on Power Systems*, vol. 16, no. 3, pp. 435–443, 2001.

[101] B. Lee, S. Kwon, J. Lee, H. Nam, J. Choo, and D. Jeon, "Fast Contingency Screening for Online Transient Stability Monitoring and Assessment of the KEPCO System," *IEE Proceedings – Generation, Transmission and Distribution*, vol. 150, no. 4, pp. 399–404, 2003.

[102] H.-D. Chiang, *Direct Methods for Stability Analysis of Electric Power Systems: Theoretical Foundation, BCU Methodologies and Applications*. Hoboken, New Jersey: John Wiley & Sons, Inc, 2011.

[103] N. Kakimoto, Y. Ohsawa, and M. Hayashi, "Transient Stability Analysis of Multimachine Power System with Field Flux Decays via Lyapunov's Direct Method," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-99, no. 5, pp. 1819–1827, 1980.

[104] T. Athay, R. Podmore, and S. Virmani, "A Practical Method for the Direct Analysis of Transient Stability," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, no. 2, pp. 573–584, 1979.

[105] M. Ribbens-Pavella and F. Evans, "Direct Methods for Studying Dynamics of Large-Scale Electric Power Systems – A Survey," *Automatica*, vol. 21, no. 1, pp. 1–21, 1985.

[106] H. Song and M. Kezunovic, "Stability Control using PEBS Method and Analytical Sensitivity of the Transient Energy Margin," in *IEEE PES Power Systems Conference and Exposition*, New York, NY, USA, October 2004.

[107] M. Benidris, N. Cai, and J. Mitra, "A Fast Transient Stability Screening and Ranking Tool," in *Power Systems Computation Conference*, Wroclaw, Poland, August 2014.

[108] T. Weckesser, H. Jóhannsson, S. Sommer, and J. Østergaard, "Investigation of the Adaptability of Transient Stability Assessment Methods to Real-Time Operation," in *2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*, Berlin, October 2012.

[109] T. L. Vu and K. Turitsyn, "A Framework for Robust Assessment of Power Grid Stability and Resiliency," *IEEE Transactions on Automatic Control*, vol. 62, no. 3, pp. 1165–1177, 2017.

[110] J. Mitra, M. Benidris, and N. Cai, "Use of Homotopy-Based Approaches in Finding Controlling Unstable Equilibrium Points in Transient Stability Analysis," in *2016 Power Systems Computation Conference (PSCC)*, Genoa, Italy, June 2016.

[111] M. Oluic, M. Ghandhari, and B. Berggren, "Methodology for Rotor Angle Transient Stability Assessment in Parameter Space," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1202–1211, 2017.

[112] T. Weckesser, H. Jóhannsson, M. Glavic, and J. Østergaard, "An Improved On-Line Contingency Screening for Power System Transient Stability Assessment," *Electric Power Components and Systems*, vol. 45, no. 8, pp. 852–863, 2017.

[113] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A Fresh Approach to Numerical Computing," *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017.

[114] H. Çakmak, H. Maass, F. Bach, and U. Kühnapfel, "A New Framework for the Analysis of Large Scale Multi-Rate Power Data," KIT Scientific Working Papers, vol. 21, Karlsruhe Institute of Technology (KIT), 2014.

[115] B. Hartmann, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Enhancing Model Interchangeability for Powerflow Studies: An Example of a New Hungarian Network Model in PowerFactory and *e*ASiMOV," in *European Conference on Modelling and Simulation, ECMS*, Budapest, Hungary, May 2017.

[116] M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Generator Model Extension for Higher Accuracy Simulation of Power System Transients in OpenModelica," in *Fourth International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, Corfu, Greece, August 2017.

[117] M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Simulation Tool for the Analysis of Unsymmetrical Power System Transients in Large Networks," in *European Conference on Modelling and Simulation, ECMS*, Wilhelmshaven, Germany, May 2018.

[118] B. Stott, "Power System Dynamic Response Calculations," *Proceedings of the IEEE*, vol. 67, no. 2, pp. 219–241, 1979.

[119] "IEEE Guide for Synchronous Generator Modeling Practices and Applications in Power System Stability Analyses," IEEE Std 1110-2002, IEEE Power Eng. Soc, New York, November 2003.

[120] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies," IEEE Std 421.5-2005, IEEE Power Eng. Soc, New York, April 2006.

[121] Task Force on Turbine-Governor Modeling, "Dynamic Models for Turbine-Governors in Power System Studies," Technical Report, IEEE Power and Energy Society, January 2013.

[122] M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Poster Abstract: Implementation of an Extended Generator subsystem in OpenModelica and Comparative Analysis with DIgSILENT PowerFactory," *Computer Science – Research and Development*, vol. 33, no. 1, pp. 253–255, 2018.

[123] P. Fritzson, *Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica*. Hoboken, NJ: Wiley, 2011.

[124] F. Milano, "Power System Analysis Toolbox Documentation for PSAT," user manual, 2008.

[125] H. Saadat, *Power system analysis*. PSA Publishing, 2010.

[126] M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Dynamic Simulation Module for Power System Transients Analysis in the *e*ASiMOV Framework," in *2017 European Modelling Symposium (EMS)*, Manchester, UK, November 2017.

[127] S. Cole, *MatDyn Version 1.2*. KU Leuven, ESAT-ELECTA, Heverlee, Belgium, http://www.esat.kuleuven.be/electa/teaching/matdyn/, February 2010.

[128] M. L. Crow, *Computational Methods for Electric Power Systems*. Boca Raton: CRC Press, 2009.

[129] S. A. Soman, S. A. Khaparde, and S. Pandit, *Computational Methods for Large Sparse Power Systems Analysis: An Object Oriented Approach*. Boston: Kluwer Academic Publishers, 2002.

[130] S. Fliscounakis, P. Panciatici, F. Capitanescu, and L. Wehenkel, "Contingency Ranking with Respect to Overloads in Very Large Power Systems Taking into Account Uncertainty, Preventive, and Corrective Actions," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4909–4917, 2013.

[131] C. Josz, S. Fliscounakis, J. Maeght, and P. Panciatici, "AC Power Flow Data in MATPOWER and QCQP Format: iTesla, RTE Snapshots, and PEGASE." http://arxiv.org/abs/1603.01533, 2016.

[132] M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Dynamic Modeling of Wind and Solar Power Generation with Grid Support for Large-Scale Integration in Power Systems," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, The Hague, NL, October 2020.

[133] M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Dynamic Modeling and Control for Assessment of Large-Scale Wind and Solar Integration in Power Systems," *IET Renewable Power Generation (In print)*, 2020.

[134] W. Xiao, *Photovoltaic Power System: Modeling, Design, and Control*. Sydney: Wiley, 2017.

[135] M. Parvez, M. Elias, N. Rahim, and N. Osman, "Current Control Techniques for Three-Phase Grid Interconnection of Renewable Power Generation Systems: A Review," *Solar Energy*, vol. 135, pp. 29–42, 2016.

[136] A. Kloos, "Modellierung einer PV-Anlage für Lastfluss- und dynamische Stromnetz-Simulationen." Bachelor's Thesis, Karlsruhe Institute of Technology, 2018.

[137] P. Kokotović, H. K. Khalil, and J. O'reilly, *Singular Perturbation Methods in Control: Analysis and Design*, vol. 25. Philadelphia: SIAM, 1999.

[138] T. Burton, N. Jenkins, D. Sharpe, and E. Bossanyi, *Wind Energy Handbook*. Oxford: Wiley, 2011.

[139] M. Brecht, "Modellierung eines Windparks für Lastfluss- und dynamische Stromnetzsimulation." Bachelor's Thesis, Karlsruhe Institute of Technology, 2018.

[140] G. Michalke, Anca D. Hansen, and T. Hartkopf, "Control Strategy of a Variable Speed Wind Turbine with Multipole Permanent Pagnet Synchronous Generator," European Wind Energy Association (EWEA), Brussels, 2007.

[141] M. Chinchilla, S. Arnaltes, and J. C. Burgos, "Control of Permanent-Magnet Generators Applied to Variable-Speed Wind-Energy Systems Connected to the Grid," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 130–135, 2006.

[142] D. Schröder, *Elektrische Antriebe – Regelung von Antriebssystemen*. Berlin, Heidelberg: Springer Vieweg, 2015.

[143] BDEW, "Generating Plants Connected to the Medium-Voltage Network," Technical Guideline, BDEW Bundesverband der Energie- und Wasserwirtschaft e.V, Berlin, June 2008.

[144] Commission Regulation (EU) 2016/631, "Establishing a Network Code on Requirements for Grid Connection of Generators," Regulations, Official Journal of the European Union, Berlin, April 2016.

[145] H. Berndt, M. Hermann, H. D. Kreye, R. Reinisch, U. Scherer, and J. Vanzetta, "Network and System Rules of the German Transmission System Operators," TransmissionCode 2007, Verband der Netzbetreiber – VDN – e.V. beim VDEW, Berlin, August 2007.

[146] "IEEE Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interfaces," IEEE Std 1547-2018 (Revision of IEEE Std 1547-2003), IEEE Standards Coordinating Committee 21, New York, April 2018.

[147] M. S. El Moursi, W. Xiao, and J. L. Kirtley, "Fault Ride Through Capability for Grid Interfacing Large Scale PV Power Plants," *IET Generation, Transmission & Distribution*, vol. 7, no. 9, pp. 1027–1036, 2013.

[148] W. Kou, D. Wei, P. Zhang, and W. Xiao, "A Direct Phase-Coordinates Approach to Fault Ride Through of Unbalanced Faults in Large-scale Photovoltaic Power Systems," *Electric Power Components and Systems*, vol. 43, no. 8-10, pp. 902–913, 2015.

[149] M. Kyesswa, P. Schmurr, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A New Julia-Based Parallel Time-Domain Simulation Algorithm for Analysis of Power System Dynamics," in *2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, Prague, Czech Republic, September 2020.

[150] M. Kyesswa, A. Murray, P. Schmurr, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Impact of Grid Partitioning Algorithms on Combined Distributed AC Optimal Power Flow and Parallel Dynamic Power Grid Simulation," *IET Generation, Transmission & Distribution (In print)*, 2020.

[151] A. Buluç, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz, "Recent Advances in Graph Partitioning," in *Algorithm Engineering: Selected Results and Surveys*, pp. 117–158, Cham: Springer International Publishing, 2016.

[152] C. Walshaw and M. Cross, "JOSTLE – Multilevel Graph Partitioning Software: An Overview," in *Mesh Partitioning Techniques and Domain Decomposition Methods*, pp. 27–58, Stirlingshire, UK: Saxe-Coburg Publications, 2007.

[153] J. Kim and I. Hwang and Y.-H. Kim and B.-R. Moon, "Genetic Approaches for Graph Partitioning: A Survey," in *13^{th} Annual Conference on Genetic and Evolutionary Computation*, Dublin, Ireland, July 2011.

[154] P. Sanders and C. Schulz, "High Quality Graph Partitioning," in *DIMACS Implementation Challenge Workshop: Graph Partitioning and Graph Clustering*, 2013.

[155] P. Sanders and C. Schulz, "Engineering Multilevel Graph Partitioning Algorithms," in *19^{th} European Symposium on Algorithms (ESA'11)*, 2011.

[156] J. Maue and P. Sanders, "Engineering Algorithms for Approximate Weighted Matching," in *International Workshop on Experimental and Efficient Algorithms*, 2007.

[157] F. Pellegrini, "Distillating Knowledge about SCOTCH," in *Combinatorial Scientific Computing*, 2009.

[158] A. Murray, M. Kyesswa, P. Schmurr, H. Çakmak, and V. Hagenmeyer, "On Grid Partitioning in AC Optimal Power Flow," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, The Hague, NL, October 2020.

[159] G. Karypis and V. Kumar, "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.

[160] C. Coffrin, R. Bent, K. Sundar, Y. Ng, and M. Lubin, "PowerModels.jl: An Open-Source Framework for Exploring Power Flow Formulations," in *2018 Power Systems Computation Conference (PSCC)*, June 2018.

[161] T. A. Davis, *UMFPACK User Guide, Version 5.6.2*. https://www.suitesparse.com, 2013.

[162] P. S. Schmurr, "Optimizing Large Power Grid Simulations with a Parallelization and Distribution Approach." Master's Thesis, Karlsruhe Institute of Technology, 2018.

[163] Steinbuch Centre for Computing (SCC), KIT, "Forschungshochleistungsrechner ForHLR II." https://www.scc.kit.edu/dienste/forhlr2.php. Accessed: 11 December 2018.

[164] M. Kyesswa, H. K. Çakmak, L. Gröll, U. Kühnapfel, and V. Hagenmeyer, "A Hybrid Analysis Approach for Transient Stability Assessment in Power Systems," in *2019 IEEE Milan PowerTech*, Milan, Italy, June 2019.

[165] M. Ribbens-Pavella and P. G. Murthy, *Transient Stability of Power Systems : Theory and Practice*. Chichester: Wiley, 1994.

[166] M. A. Pai and P. W. Sauer, "Stability Analysis of Power Systems by Lyapunov's Direct Method," *IEEE Control Systems Magazine*, vol. 9, no. 1, pp. 23–27, 1989.

[167] HD. Chiang and H. Li and J. Tong and Y.Tada, "On-Line Transient Stability Screening of a Practical 14,500-Bus Power System: Methodology and Evaluations," in *High Performance Computing in Power and Energy Systems*, pp. 335–358, Springer, Berlin, Heidelberg, 2013.

[168] F. Dorfler and F. Bullo, "Kron Reduction of Graphs with Applications to Electrical Networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 1, pp. 150–163, 2013.

[169] W. W. Hager, "Updating the Inverse of a Matrix," *SIAM Review*, vol. 31, no. 2, pp. 221–239, 1989.

[170] O. Alsac, B. Stott, and W. F. Tinney, "Sparsity-Oriented Compensation Methods for Modified Network Solutions," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-102, no. 5, pp. 1050–1060, 1983.

[171] J. Bezanson, J. Nash, and K. Pamnany, "Announcing Composable Multi-Threaded Parallelism in Julia." https://julialang.org/blog/2019/07/multithreading. Accessed: 18 September 2019.

[172] Center for Intelligent Systems and Networks, *Dynamic IEEE Test Systems*. University of Cyprus, http://www.kios.ucy.ac.cy/testsystems/.

List of Publications

Journal Articles

[1] M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Poster Abstract: Implementation of an Extended Generator subsystem in OpenModelica and Comparative Analysis with DIgSILENT PowerFactory," *Computer Science – Research and Development*, vol. 33, no. 1, pp. 253–255, 2018.

[2] H. Çakmak, A. Erdmann, M. Kyesswa, U. Kühnapfel, and V. Hagenmeyer, "A New Distributed Co-simulation Architecture For Multi-Physics Based Energy Systems Integration," *At-Automatisierungstechnik*, vol. 67, no. 11, pp. 972–983, 2019.

[3] M. Kyesswa, A. Murray, P. Schmurr, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Impact of Grid Partitioning Algorithms on Combined Distributed AC Optimal Power Flow and Parallel Dynamic Power Grid Simulation," *IET Generation, Transmission & Distribution (In print)*, 2020.

[4] M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Dynamic Modeling and Control for Assessment of Large-Scale Wind and Solar Integration in Power Systems," *IET Renewable Power Generation (In print)*, 2020.

Conference Articles

[1] M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Generator Model Extension for Higher Accuracy Simulation of Power System Transients in OpenModelica," in *4^{th} International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, Corfu, Greece, pp. 44–50, August 2017.

[2] M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Dynamic Simulation Module for Power System Transients Analysis in the *e*ASiMOV Framework," in *2017 European Modelling Symposium (EMS)*, Manchester, UK, pp. 157–162, November 2017.

[3] M. Kyesswa, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A Matlab-Based Simulation Tool for the Analysis of Unsymmetrical Power System Transients in Large Networks," in *European Conference on Modelling and Simulation (ECMS)*, Wilhelmshaven, Germany, pp. 246–253, May 2018.

[4] M. Kyesswa, H. K. Çakmak, L. Gröll, U. Kühnapfel, and V. Hagenmeyer, "A Hybrid Analysis Approach for Transient Stability Assessment in Power Systems," in *2019 IEEE Milan PowerTech*, Milan, Italy, pp. 1–6, June 2019.

[5] M. Kyesswa, P. Schmurr, H. K. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "A New Julia-Based Parallel Time-Domain Simulation Algorithm for Analysis of Power System Dynamics," in *2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, Prague, Czech Republic, pp. 16–24, September 2020.

[6] A. Murray, M. Kyesswa, P. Schmurr, H. Çakmak, and V. Hagenmeyer, "On Grid Partitioning in AC Optimal Power Flow," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, The Hague, NL, pp. 524–528, October 2020.

[7] M. Kyesswa, H. Çakmak, U. Kühnapfel, and V. Hagenmeyer, "Dynamic Modeling of Wind and Solar Power Generation with Grid Support for Large-Scale Integration in Power Systems," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, The Hague, NL, pp. 569–573, October 2020.