



Enhanced State Estimation by Advanced Substation Monitoring

Final Project Report

Power Systems Engineering Research Center

*A National Science Foundation
Industry/University Cooperative Research Center
since 1996*





Power Systems Engineering Research Center

**Enhanced State Estimation
by Advanced Substation Monitoring**

Final Project Report

Project Team

Ali Abur and Mladen Kezunovic
Texas A&M University

Sakis Meliopoulos
Georgia Institute of Technology

PSERC Publication 02-48

November 2002

Information about this Project

For information about this project contact:

A. Abur
Professor
Texas A&M University
Department of Electrical Engineering
College Station, TX 77843-3128
Phone: 979-845-1493
Fax: 979-845-9887
Email: abur@tamu.edu

Power Systems Engineering Research Center

This is a project report from the Power Systems Engineering Research Center (PSERC). PSERC is a multi-university Center conducting research on challenges facing a restructuring electric power industry and educating the next generation of power engineers. More information about PSERC can be found at the Center's website:
<http://www.pserc.wisc.edu>.

For additional information, contact:

Power Systems Engineering Research Center
Cornell University
428 Phillips Hall
Ithaca, New York 14853
Phone: 607-255-5601
Fax: 607-255-8871

Notice Concerning Copyright Material

PSERC members are given permission to copy without fee all or part of this publication for internal use if appropriate attribution is given to this document as the source material. This report is available for downloading from the PSERC website.

Acknowledgements

The work described in this report was sponsored by the Power Systems Engineering Research Center (PSERC). We express our appreciation for the support provided by PSERC's industrial members and by the National Science Foundation under grant NSF EEC-0002917 received under the Industry / University Cooperative Research Center program.

The industry advisors for the project were Mani Subramanian, ABB Network Management; Don Sevcik, CenterPoint Energy; and Bruce Dietzman, Oncor. Their suggestions and contributions to the work are appreciated.

Executive Summary

Power system substations are becoming equipped with intelligent electronic devices (IEDs) that are primarily used for protection and relaying functions as well as for power quality monitoring. The advanced communication and computational capabilities provided by these devices can be exploited to improve energy management system (EMS) applications.

This project is concerned about one such application: the monitoring of power systems during normal operating conditions. Conventional monitoring involves scanning the measurements from various substations in the system and estimating the system state using a state estimator at the EMS control center. While state estimators can detect and identify most types of analog measurement errors, they are still vulnerable to errors in system topology. These errors show up when the assumed status of circuit breakers and switches does not coincide with their true status. This project investigates solutions to the detection, identification and elimination of such errors via the help of substation IEDs.

One of the reasons why certain topological errors go undetected is that some measurements at the substation level cannot be used by the state estimator. This is because the overall system model does not include the detailed representation of substation configurations. In this project, these redundant substation-level measurements are integrated into the state estimator to improve topology error identification, error rejection and statistical robustness features. Furthermore, other error sources that are not topological, such as errors due to phase imbalances, are analyzed to determine their influence on the state estimation results.

This project's results suggest that processing measurements at the substation level before sending them to the control center can improve state estimator performance. In addition, availability of detailed substation measurements can facilitate topology error identification by the state estimator executed at the control-center level. These benefits are demonstrated under various operating scenarios using prototype software developed as part of the project.

Table of Contents

1. Introduction.....	1
1.1 Problem Definition	1
1.2 Objectives and Scope.....	1
1.3 Summary of Results.....	1
1.4 Organization of the Report	2
2. Substation Level Data Processing	3
2.1 Substation Modeling.....	3
2.1.1 Equivalent Source	5
2.1.2 Switch Element	6
2.1.3 Analog Measurement Blocks	7
2.1.4 Triggering Blocks.....	8
2.2 Data Processing and Inconsistency Checks	9
2.2.1 Data Preparation.....	9
2.2.2 Description of the Measurement Placement and Topology.....	9
2.2.3 Processing and Checking	10
2.3 References.....	20
3. State Estimation and Topology Errors.....	21
3.1 Formulation of State Estimation and Bad Data Detection.....	21
3.1.1 WLS Methods	21
3.1.2 WLAV Methods	22
3.1.3 Bad Data Detection/Identification.....	23
3.2 Topology Error Detection and Identification.....	24
3.3 State Estimation Using the Substation Model	24
3.4 Modeling of the Substation.....	26
3.5 Formulation of the Two-Stage State Estimation.....	31
3.5.1 The First-Stage State Estimator.....	31
3.5.2 Suspect Substation Identification.....	31
3.5.3 Second-Stage State Estimation and Correction of Topology Errors	32
3.6 New Suspect Substation Identification Methods.....	35
3.6.1 Considerations for Improving the Identification Performance	35
3.6.2 Comprehensive Identification Methods	37
3.7 Library of Topology Error Scenarios.....	38
3.7.1 Type 1: Merged Bus	39
3.7.2 Type 2: Split Bus.....	39
3.7.3 Type 3: Line Outage.....	39
3.7.4 Type 4: Connected Line	39

Table of Contents (continued)

3.8 Simulation Results for Two-Stage State Estimation	41
3.8.1 Case 1: Bus Split Case	41
3.8.2 Case 2: Line Outage Case	45
3.9 Simulation Results for Suspect Substation Identification Methods	49
3.10 References	52
4. Three Phase State Estimation-Sensitivity Study	54
4.1 Shortcoming of the Single Phase State Estimation	54
4.2 Possible State Estimation Methods	54
4.2.1 Estimate 1 (Three-phase)	54
4.2.2 Estimate 2 (Single-phase)	54
4.2.3 Estimate 3 (Positive sequence)	55
4.3 Three-Phase State Estimation Formulation and Modeling	55
4.3.1 Three Phase Transmission Lines	55
4.3.2 Three-Phase Loads and Generators	56
4.3.3 Transformers	56
4.3.4 Bus Shunts	57
4.4 Sensitivity Study: Effect of Neglecting 3-Phase Unbalances	57
4.4.1 Study Cases	57
4.4.2 Investigation Methodology	58
4.4.3 Sensitivity Indices	59
4.5 Simulation Results	59
4.5.1 Non-Transposed Cases	60
4.5.2 Unbalanced Cases	61
4.5.3 Conclusion of the Sensitivity Studies	65
4.6 References	65
5. Graphical Interface for the Two-Stage State Estimation	66
5.1 Auxiliary Substation Window	67
5.2 Editing the Substation Model	67
5.2.1 Creation of the Substation Model from Scratch	67
5.2.2 Creation of a Substation Model Using the Library	69
5.3 Setting the Bus's Property for Related Substation Model	70
5.4 Setting the Parameters of the Substation Model	71
5.4.1 Property of a Node	71
5.4.2 Property of a Branch	73

Table of Contents (continued)

5.5 The Navigator Window and Printing Function.....	74
5.5.1 Navigator Window	74
5.5.2 Printing Function.....	74
5.6 Automatic Generation of Substation Measurements	75
5.7 Procedure to Test the Two-Stage State Estimation	75
5.8 Examples.....	76
5.9 References.....	82
6. Graphical User Interface for the Substation Processor	83
6.1 Graphical User Interface (GUI).....	83
6.2 User's Guide on Running the Simulation	85
6.3 Modes of Operation	92
6.4 Software Components and Installation	95
6.5 Example of Software Outputs.....	96
7. Conclusions and Future Work.....	100
7.1 Project Accomplishments	100
7.2 Conclusions and Future Work.....	100

1. Introduction

1.1 Problem Definition

This project is concerned with the monitoring of power systems during normal operating conditions. Conventional monitoring involves scanning the measurements from various substations in the system and estimation of the system state using a state estimator at the EMS control centers. While the state estimators can detect and identify most types of analog measurement errors, they are still vulnerable to errors in the topology of the system. These errors show up when the assumed status of the circuit breakers and switches do not coincide with their true status. This project investigates solutions to the detection, identification and elimination of such errors.

1.2 Objectives and Scope

In order to solve the problem of topology errors, the project investigates the integration of new types of analog and digital (contact status) measurements, which are made available at the substations via the new generation of digital relays and other IEDs, into various state estimation functions in energy management systems. The reliability of existing state estimators can be improved if a robust method can be developed for detecting, identifying and eliminating topology errors. One of the reasons why certain topological errors go undetected is that some of the measurements that exist at the substation level cannot be utilized by the state estimators since the overall system model does not include detailed representation of the substation configurations. This project integrates these redundant measurements at the substation level into the state estimator and attempts to improve its topology error identification, error rejection and statistical robustness features. Furthermore, other sources of error that are not topological, such as those due to the phase imbalances between the three phases, are analyzed to determine their influence on the state estimation results.

1.3 Summary of Results

The following project tasks are carried out and the results are presented in this report:

- A two stage state estimator which can detect and identify substation topology errors, is developed, implemented and tested on IEEE test systems.
- Substation level data processing software is developed in Matlab. Simulations are carried out to create the full three phase detailed model and the corresponding measurements from a typical substation.
- A three phase state estimation program is developed.

- A sensitivity study is carried out to determine the influence of network load unbalances and non-transposed transmission lines on the positive sequence state estimate. Results are documented and presented in this report.
- A new and improved procedure for selecting suspect substations after the first-stage state estimation is developed. This procedure, along with other alternative procedures, are comparatively evaluated using a customized library of cases specifically created for this purpose. The library contains fifty cases corresponding to four different types of topology error scenarios and is based on the IEEE 30 bus test system.
- A graphical user interface that can allow substation model creation, automatic generation of consistent substation measurements and integrating them into the detailed state estimation model in the second stage, is implemented and tested.

1.4 Organization of the Report

This is the final report for the project and contains seven sections. The substation level data processing program description is given in section 2. A detailed description of the state estimation formulation, network modeling and the developed two stage state estimator is given in section 3. This is followed by section 4 that presents the sensitivity study for the three phase estimator. The detailed description of the graphical user interface development for the two stage estimator is given in section 5. Section 6 presents the graphical user interface for the substation measurement processor developed in the Matlab environment. The report ends with a summary of project's main accomplishments and suggestions for future work.

2. Substation Level Data Processing

2.1 Substation Modeling

All the modeling is performed in SIMULINK [1]. Detailed single-phase substation model is developed using standard power blockset elements. The model is invoked and controlled by the software. It is executed in the background and there is no need for users to make any actions directly on the model. Data flow between the model and the rest of the software is accomplished automatically and since SSES offers comprehensive Graphical User Interface (GUI), parameters and topology changes are controlled by users through the interface.

The substation model is designed to constantly generate raw data that simulate information obtained from the physical substation in reality. This data is communicated in predetermined intervals for further processing by other software components. Our substation model represents one typical substation layout [3, 4] both graphically and functionally. Several major blocks are implemented in order to model important elements and functions in the substation.

Schematic of the chosen substation layout is shown in Fig. 2-1. It consists of two busbars with breaker-and-a-half type connection with the transmission lines and double-bus-double-breaker type connection with the load.

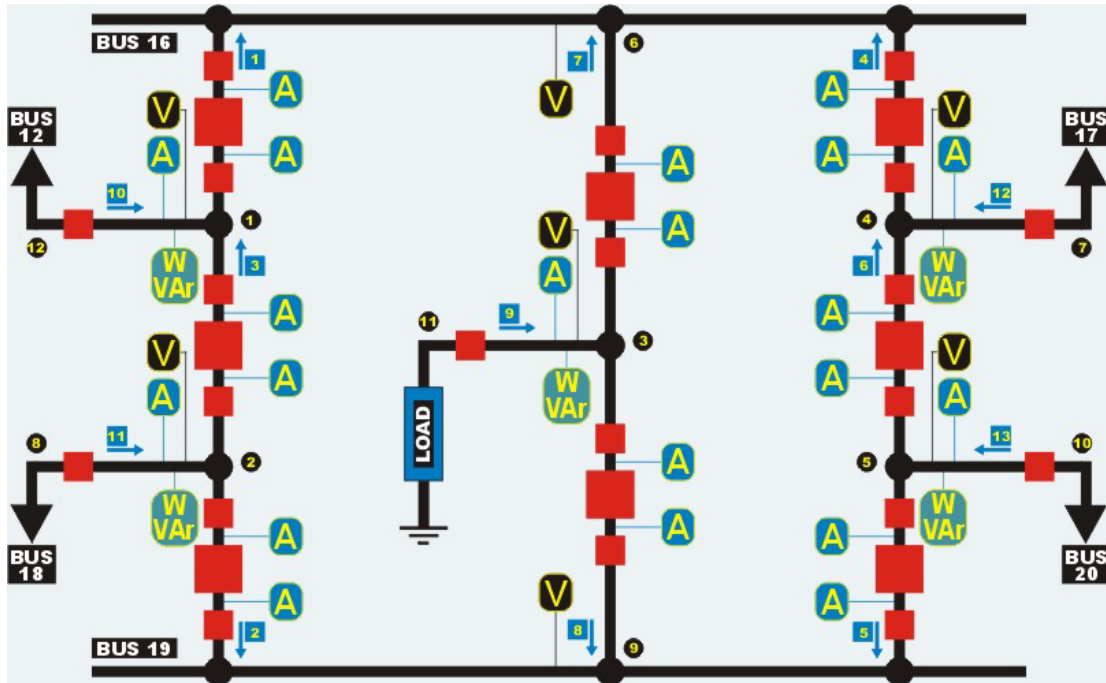


Fig. 2-1. One line diagram of a substation layout with measurement device placements

The layout of the corresponding model built in SIMULINK is shown in Fig. 2-2.

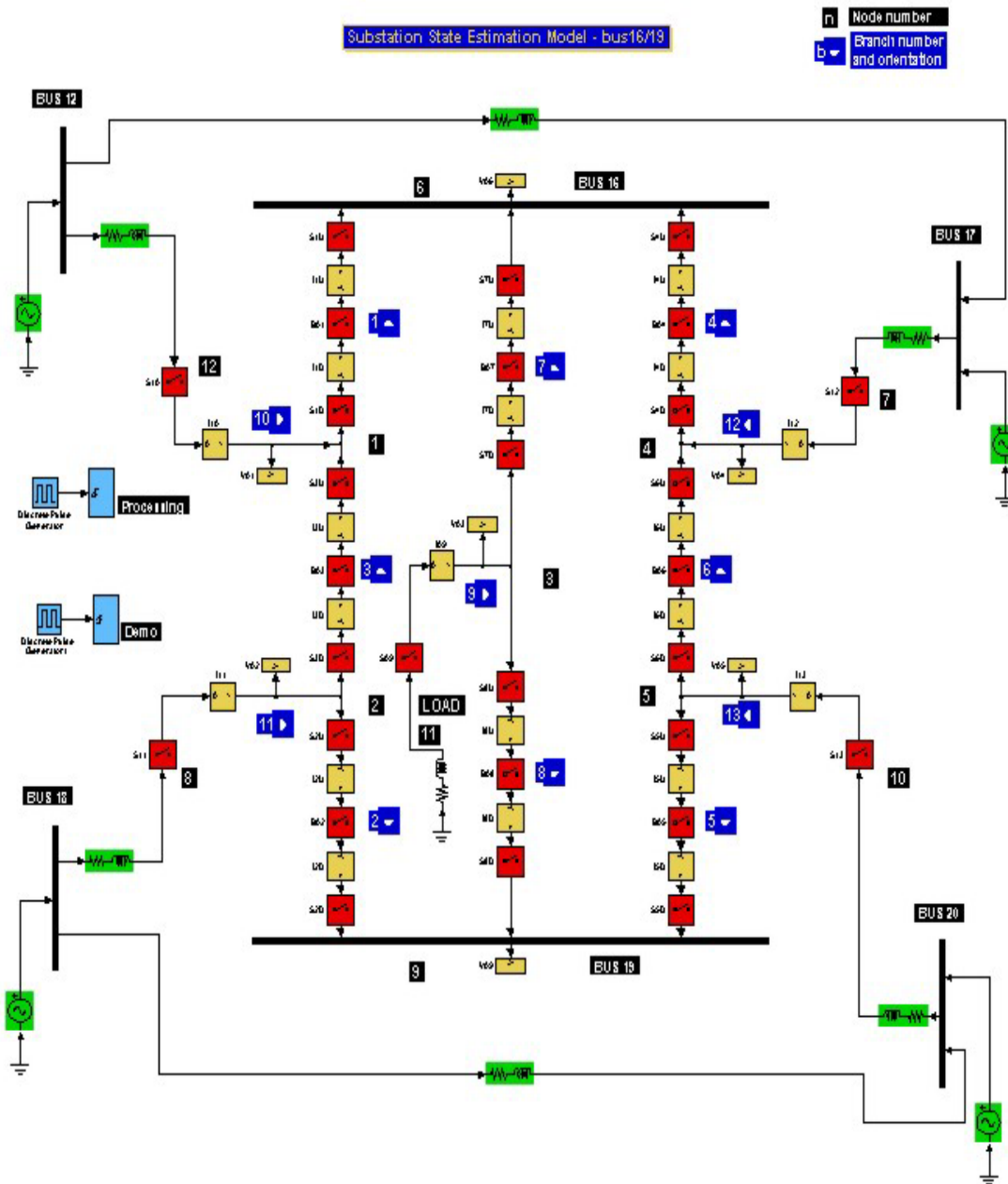


Fig. 2-2. SIMULINK-based model for Substation State Estimation Software

Four main groups of SIMULINK blocks are used in the model, each colored differently. The green blocks represent the equivalent sources (generators and impedances) where the rest of the network is equivalenced. The red blocks are switch elements (circuit breakers or disconnect switches), the yellow blocks represent the measuring units (either current or voltage), and blue blocks are triggering units that control data exchange rate between software components.

The distribution (location) of analog and digital measurements in the substation model is determined considering following rules:

- Each circuit breaker has two current measurements (one at each side).
- Each transmission line has one current measurement, one voltage measurement, and calculated active and reactive power measurements.
- Busbars have one voltage measurement.
- All switch elements have contact status measurements.

All measurements are single-phase measurements. Type and placement of analog measurement devices within the model is shown in Fig. 2-1. Also, circuit breakers and switches are represented with red squares and they all assume that digital status measurement is provided. Designation of nodes, branches and power elements in Figures 1 and 2 is consistent throughout the software.

2.1.1 Equivalent Source

Equivalent source block (green color in Fig. 2-2) consists of an equivalent ideal generator and corresponding equivalent impedances toward neighboring substations in the network. Fig. 2-3 shows the equivalent source block layout.

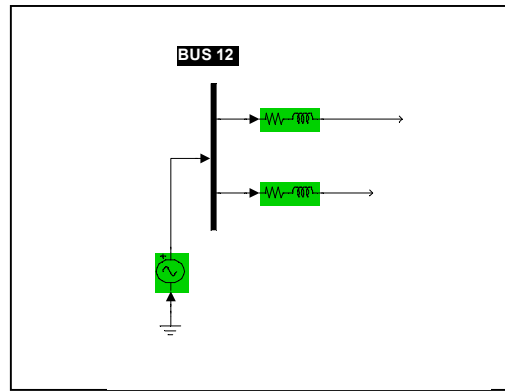


Fig. 2-3. Equivalent source block layout (green color)

Generator and impedance values are calculated by the main software routine based on the network equivalent data. The generator block receives values for voltage magnitude and phase angle parameters whereas its frequency is set to 60 Hz. Impedances (resistances and inductances) are also obtained from the main software routine. As it can be seen in Fig. 2-2, there are some additional connections between some of the equivalent source blocks which model closed loops in the rest of the power network.

2.1.2 Switch Element

Switch elements (red blocks in Fig. 2-2) are modeled as controllable switches. They can be opened or closed by the user through the GUI. Status can be changed at any time during the simulation. A discrete pulse generator controlled by the main software routine determines the times when the switch element status is captured and transferred for processing. Switch element block layout is shown in Fig. 2-4.

Circuit breakers and disconnect switches are modeled with the same switch element block. Their parameters (parallel resistance and series inductance) are obtained from the main software routine. All switch elements have one resistor (with high value of resistance) in parallel and one inductor (with low value of inductance) in series. Those additional elements are necessary for the SIMULINK software set-up to run the model properly. (Switch elements are modeled as current generators and their series connection is not allowed if they are stand-alone elements.)

Status of the switch elements is captured from appropriate control blocks. Separately written subroutines take care that status of each switch block is being properly reported to the software processing routine. This information is later used in substation topology determination. The status can only have two different values: “1” for closed and “0” for open.

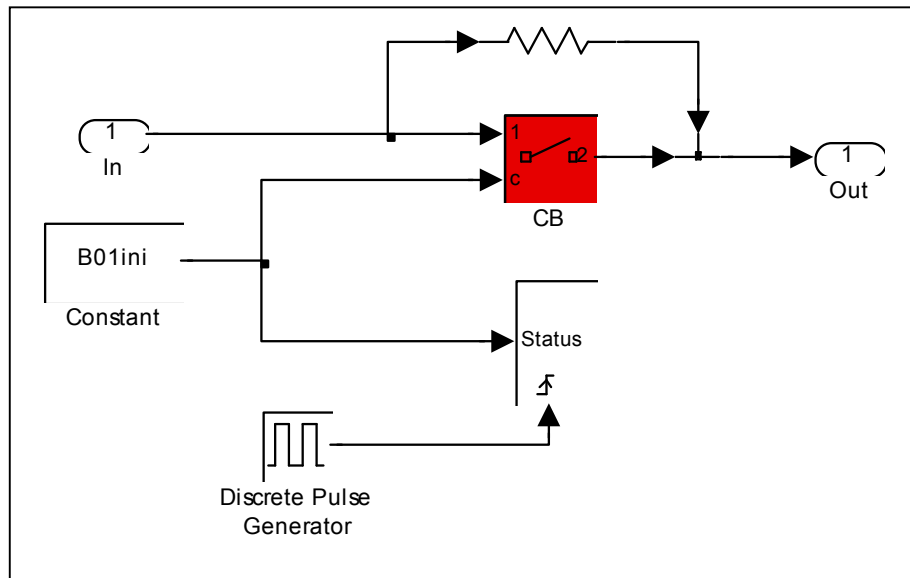


Fig. 2-4. Switch element block (red color)

2.1.3 Analog Measurement Blocks

There are two types of analog measurement blocks: current and voltage measurements (yellow blocks in Fig. 2-2). (Layouts are shown in Figures 5 and 6.) They are structurally the same, except that voltage measurements have only an input (parallel connection) whereas current measurements have both inputs and outputs (series connection).

Continuous (time domain) signals obtained from current and voltage measurement blocks (yellow on Figures 5 and 6) are led to Fourier Analyzers (colored cyan on Figures 5 and 6). A Fourier transform is performed over a running window of one cycle of fundamental frequency so that the magnitude and phase angle are extracted from the continuous signal being measured [5, 6]. Those two parameters completely determine phasors of measured electrical quantities.

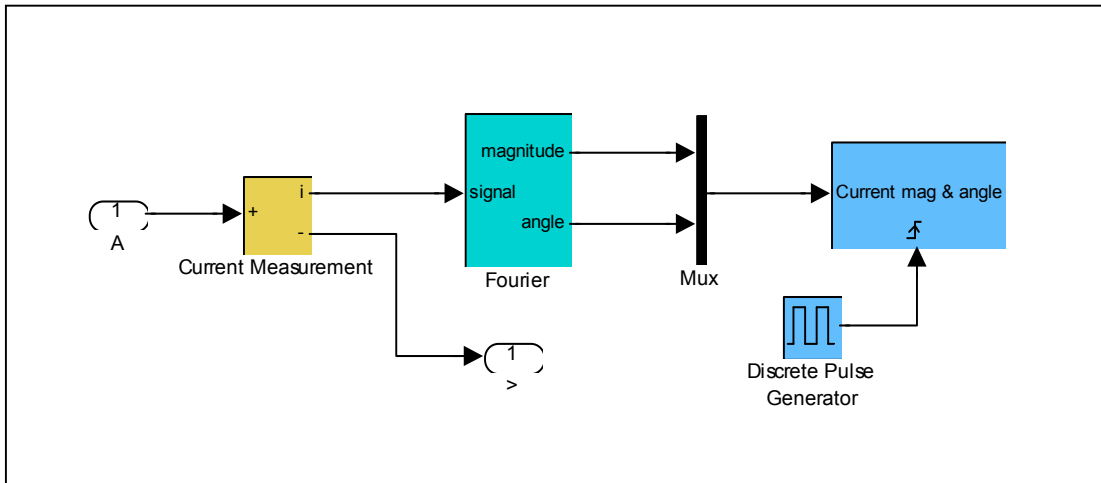


Fig. 2-5. Current measurement block (yellow color)

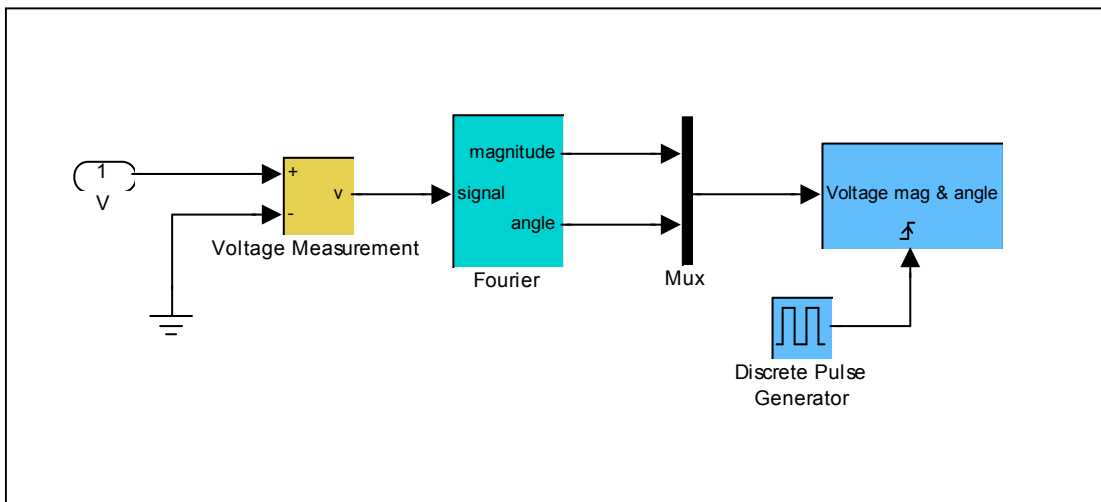


Fig. 2-6. Voltage measurement block (yellow color)

In the same manner as with switch element statuses, discrete pulse generators control current and voltage phasor transfers. Since all the discrete pulse generators are controlled by the main software routine, their operation is synchronized. Thus, all the analog and status measurements throughout the model are captured in the same moments in time. This is very important since it facilitates operations on phasors. It is assumed that the real substation set-up would provide means to achieve the same outcome with real phasor measurements.

2.1.4 Triggering Blocks

Triggering blocks (blue color blocks in Fig. 2-2) are important elements in the process of providing the captured measurements to the main software routine. Their purpose is to call processing and demo subroutines that manage utilization of measurements from the SIMULINK model. The major idea is implemented through simple block structures shown in Fig. 2-7.

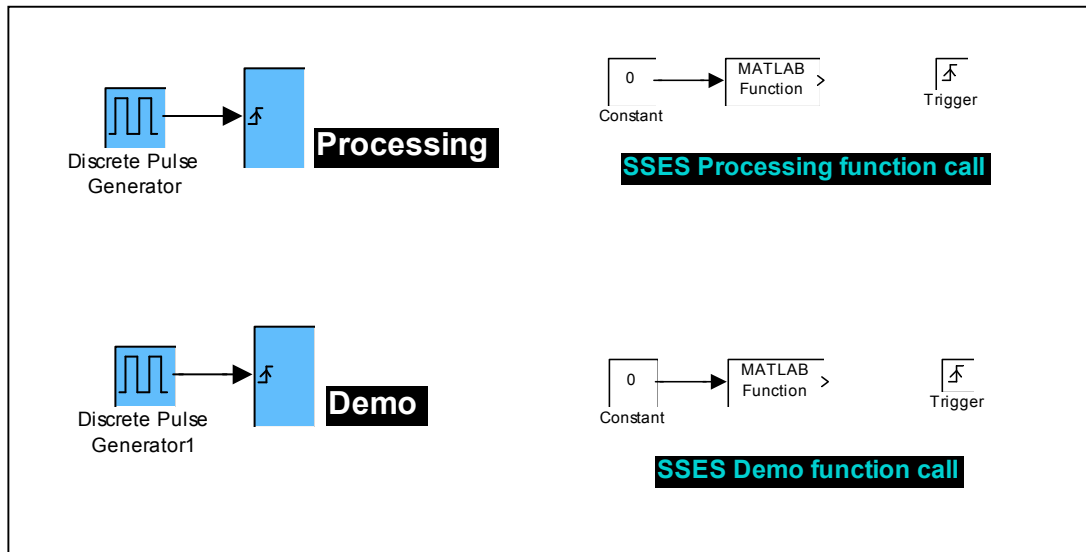


Fig. 2-7. Triggering blocks (blue color)

Both processing and demo function blocks are triggered by discrete pulse generators that determine time instants of their call. This results in the processing of simulated substation data while the simulation of the substation model is running.

Discrete pulse generators connected to all measurement blocks trigger at the same time and the substation data snapshot is memorized. After a short delay, discrete pulse generators that are connected to the processing and demo blocks trigger execution of corresponding subroutines. Thus, a data snapshot is being processed shortly after being generated.

2.2 Data Processing and Inconsistency Checks

Several routines take care of the data snapshots created by the substation model. The most important among them is the processing routine. The task of the processing routine is to acquire data of the latest snapshot, perform preprocessing, apply implemented checking algorithms, and generate outputs. The processing routine is written in MATLAB programming language.

The processing routine actually simulates the substation computer that collects data from the switchyard and prepares it for further communication to the main state estimation computer. It filters data and yields processed information in a form understandable to the higher level computer (level of the whole power system). Since the processing is accomplished at the level of substation, it is called substation state estimation.

The processing routine performs processing and consistency checking of collected snapshot data. It applies several algorithms that will be described later.

2.2.1 Data Preparation

First task of the processing routine is to prepare data for further processing. The substation model generates analog measurement and topology data snapshots. This is just a part of data set that is used for processing. Active and reactive powers are computed and supplemented to the set. They are obtained by multiplying corresponding voltage and current phasors. In real substations, the power data may be obtained either directly from the transducers or through calculation. The substation model can be modified to generate power measurements as well, but in order to avoid complexity, the power measurements are accomplished through preprocessing rather than through simulation.

As it was mentioned earlier, each measurement has its own subroutine that takes care of storing corresponding values within the snapshot. Those subroutines also perform part of the preprocessing task before a value is stored. First, the phasor angles are converted from degrees into radians. Second, phasors are calculated as complex numbers from their polar components.

2.2.2 Description of the Measurement Placement and Topology

In order to perform processing of data, the topology of modeled substation as well as the placement of measurements (analog and digital) need to be described. Topology is described by listing the node numbers (consecutive numbers starting from one) and their corresponding classifications. Nodes can be busbars (classification 1), external nodes (2) or internal nodes (3). In addition, branches are listed in a separate matrix that contains the branch number “from” node and “to” node of each branch. Branch orientation is also determined by the sequence of terminal nodes. These descriptions reflect the part of the data that does not change for the modeled substation.

Part of the data that can change from snapshot to snapshot is prepared by the processing routine. Since the processing routine is called after each snapshot has been generated, it was natural to accomplish this task within this routine. The substation model provides branch current measurements, voltage measurements for some nodes, and status of all switch elements. The processing routine needs to know what are the available measurements, what is the value of a particular measurement, and where the associated instruments are located within the substation. Therefore, four matrices are defined for this purpose: branches with current measurements, nodes with voltage measurements, branches with calculated power flows, and branches with switch elements and their status. All these matrices are three dimensional since the values for all the quantities can differ in time (for different snapshots).

The matrix that describes statuses of switch elements is created differently depending on the mode the software is operating in. Modes of operation will be discussed later, but it can be said now that statuses can be obtained either from the substation model (corresponds to data obtained from the switchyard in reality) or from the user interface (when in Bad Data mode that simulates erroneous status acquisition).

2.2.3 Processing and Checking

This is the core of the processing routine. Processing and checking is performed for each snapshot separately. Also, each snapshot is completely processed before another one arrives. This corresponds to the way the data is processed by the substation computer (real time processing). For the purpose of time-series checking, data from previous snapshots is stored in computer memory and the snapshot history is readily available.

Five different processing and checking algorithms are implemented in the processing routine. All will be described next.

- **Double current measurements - consistency check**

Some branches have two measurements of current (branches with circuit breakers with two current transformers in their bushings). One redundant measurement of current can also be obtained from a digital relay or some other intelligent electronic device (IED) monitoring the branch [7]. In these cases, it needs to be decided what is the value of the current in the branch. This is easy if both measurements agree, but becomes a little more difficult when a discrepancy exists.

This algorithm calculates one value for the branch current based on both measurements and performs consistency check at the same time. Ideally, both values should be almost equal. Algorithm flowchart is shown in Fig. 2-8.

The algorithm treats all the branches. It determines first if there is a redundant measurement of current in the branch. If there is only one measurement of current, then that measurement is assigned for the branch current and the rest of the logic is skipped.

In branches where redundant measurement exists, consistency checking is performed. The criterion is that the absolute value of the difference between the measured phasors should be less than a certain percent of the absolute value of larger measurement. Percent is determined by the variable *MADMDis* (Maximum Allowable Double Measurement Discrepancy), which is initiated in the software main routine (with a default value of 0.0001). Its value can be changed through the user interface. The assumption is that there will be no discrepancy in a phase angle without a discrepancy in magnitude.

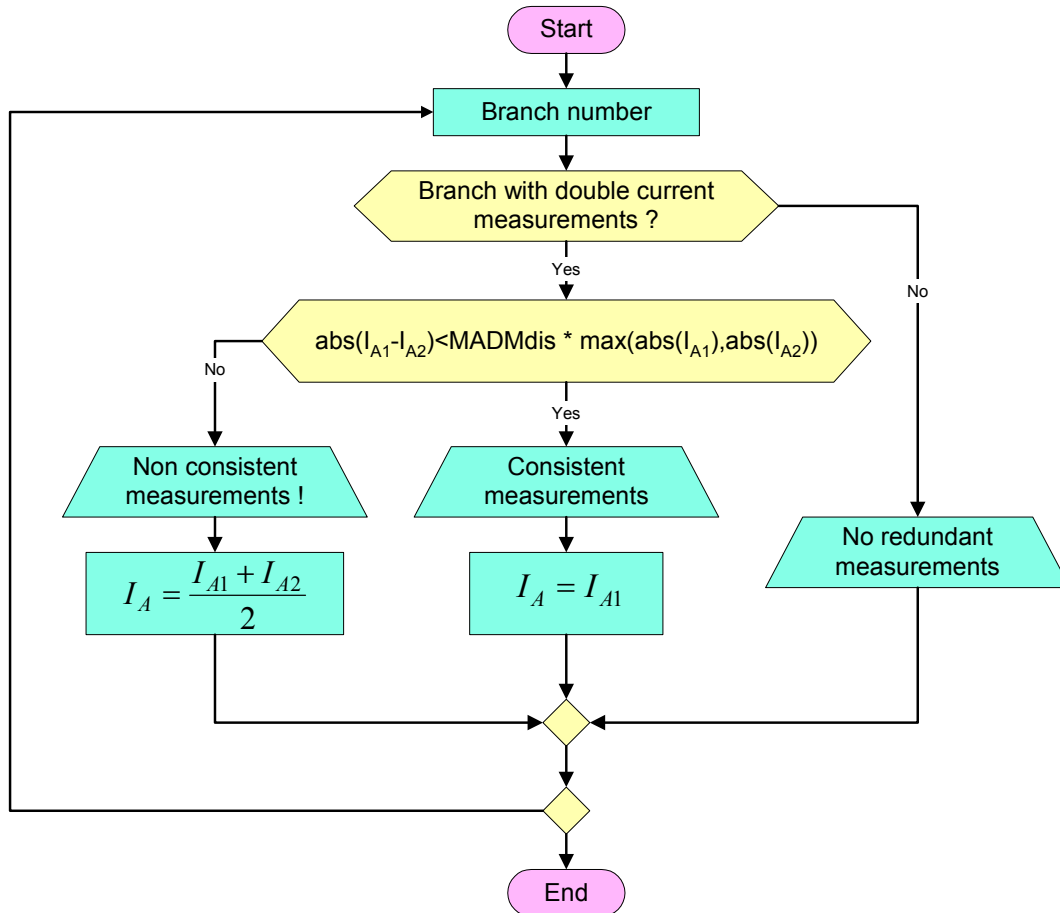





Fig. 2-8. Algorithm for branch currents consistency checking

If the criterion is satisfied, the icon between two A-meter symbols (on the GUI) is set to the vertical equality sign  to note consistency. The current in the branch is assigned to be one of the current measurements (since they are the same to the precision determined by *MADMDis* variable).

In the case when the current criterion is not satisfied, the icon between two A-meter symbols (on the GUI) is set to the  sign to note inconsistency. The GUI processing report is also generated (e.g. ). The current in the branch is determined as an average value of two current measurements. Alternative for this conclusion would be to mark the current in this branch and upon additional checks to reject one of the measurements (as bad data).

- **First Kirchhoff's law consistency check**

This type of check can be performed for all nodes where three or more branches meet and the measurements of current exist in all branches. The algorithm is shown in Fig. 2-9.

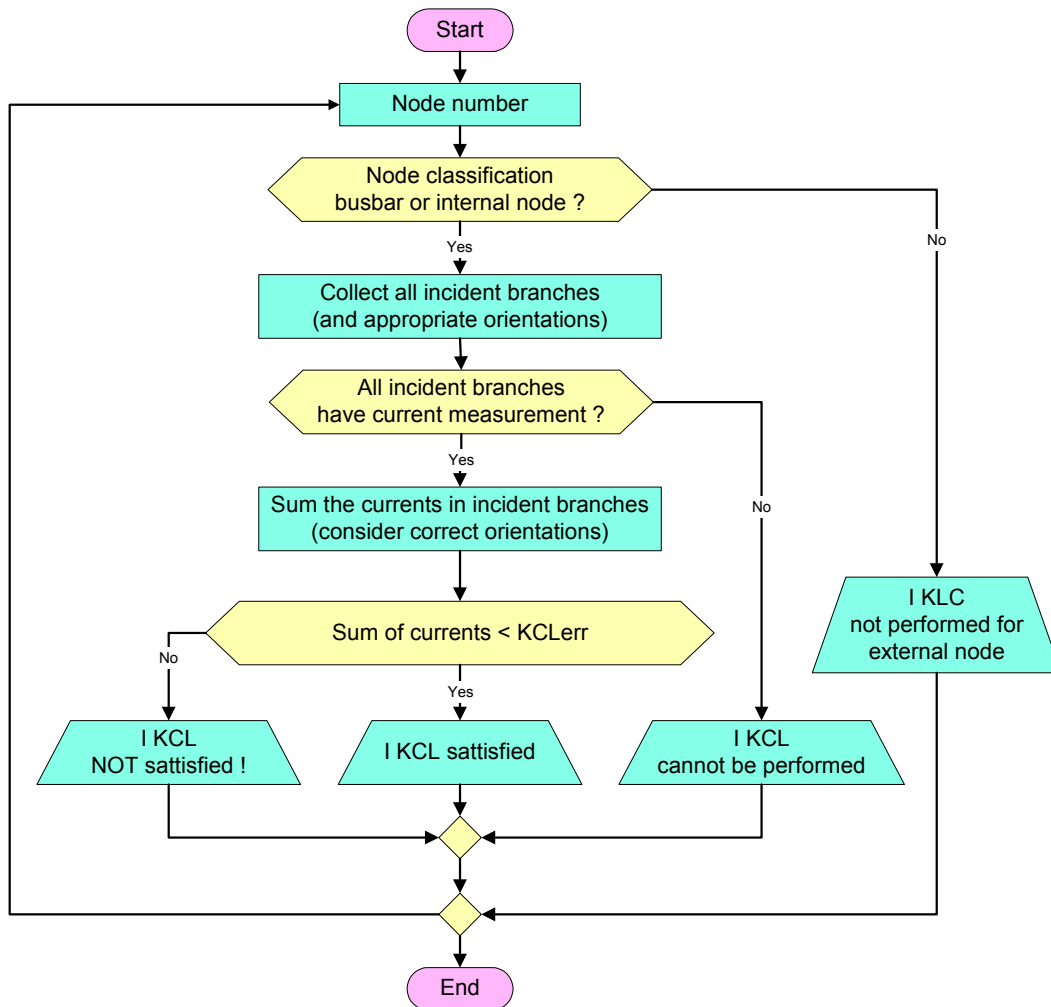





Fig. 2-9. Algorithm for Kirchhoff's current law check

The algorithm treats all nodes and first checks the node classification. Only busbars and internal nodes are taken into consideration. The first Kirchhoff's law is not performed for external nodes since they split one branch (that connects two substations) into two parts. Current measurement in the remote part of the branch is usually not accessible and also those nodes are incident with only two branch parts, which is not enough to perform the check.

For nodes with classification 1 (busbars) or 3 (internal nodes), the incident branches need to be determined and a check needs to be performed to determine if all of them are equipped with the units measuring currents. If this is not the case, the node is skipped since the first Kirchhoff's law cannot be checked because some branch current measurements are missing.

For the internal nodes, there is enough information to check Kirchhoff's Current Law (KCL). Currents in the node incident branches are summed considering orientation of each current (leaving or entering the node). Then, the first Kirchhoff's law condition is checked. Ideally, the sum of currents should be zero, but the processing routine allows the existence of certain error. This error is defined with variable **KCLerr**, whose value is initiated in the software main routine (default value of 0.0001) and can be changed through the user interface.

If the KCL condition is satisfied, the icon next to the node (on the GUI) is set to the  sign to note consistency. In the opposite case, the icon is set to the  sign to note inconsistency. In addition, the GUI processing report is generated (e.g. ). After reporting an inconsistency no further investigation is conducted to determine why the sum of currents deviates from zero.

- **Branch status determination**

Determination of branch statuses is accomplished considering all switch elements in particular branches. The configuration of switch elements for the chosen substation layout is either one disconnect switch, or one circuit breaker and two disconnect switches in the branch. The algorithm shown in Fig. 2-10 treats all branches and counts if there is one or three switch elements in any branch.

If the branch has only one switch element, that one is a disconnect switch. Branch status is determined based on the status of that switch; i.e., the branch status is the same as the switch status.

For branches with three switch elements, status is determined to be “1” (closed) only if all switch elements in the branch are “1”. In other words, if only one switch element in the branch is “0” (opened), the branch status will be determined as “0”.

This algorithm performs only determination of branch statuses. No check is done and therefore no processing report is generated at this time. The algorithm takes care to set each switch element icon (on the GUI) to an appropriate value that will reflect its status: **0** for opened or **1** for closed. As it will be described next, the branch status determined here is not definite since the consistency check algorithm, that is applied later, can alter the branch status after it considers more comprehensive data.

The substation model does not contain a representation of ground switches. Those can be added and the branch status algorithm can be expanded appropriately to include their statuses after additional data consistency checks and improved topology determination. Statuses of ground switch elements are not reported to the system state estimation computer and they can be used only locally (at the level of substation).

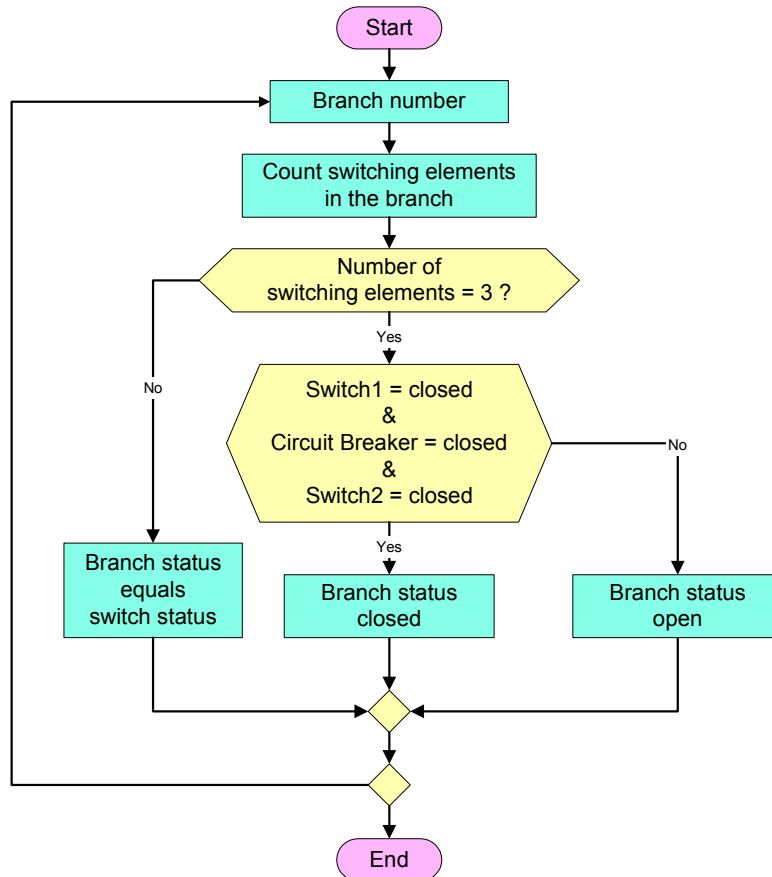


Fig. 2-10. Algorithm for branch status determination

A ground switch element that is closed can be used to infer the node voltage it is connected to (i.e., that voltage should be zero). In addition, no branch should be in status “1” if it connects such node with other one that is at the rated voltage. Ground switches are usually closed when some maintenance work is in progress and their status can guide the switching sequence control program to follow the corresponding path in analyzing and predicting topology in the substation.

- **Consistency check of branch current values and statuses**

The following algorithm is developed in order to perform a consistency check between branch current values and branch statuses (topology data). It determines the correct switch element status based on additional information about the branch current. This is the most complex algorithm applied in the processing routine due to several possible combinations of branch current values and branch statuses. Possible combinations reflect different situations and have a distinct impact on the conclusion. The algorithm flowchart is shown in Fig. 2-11.

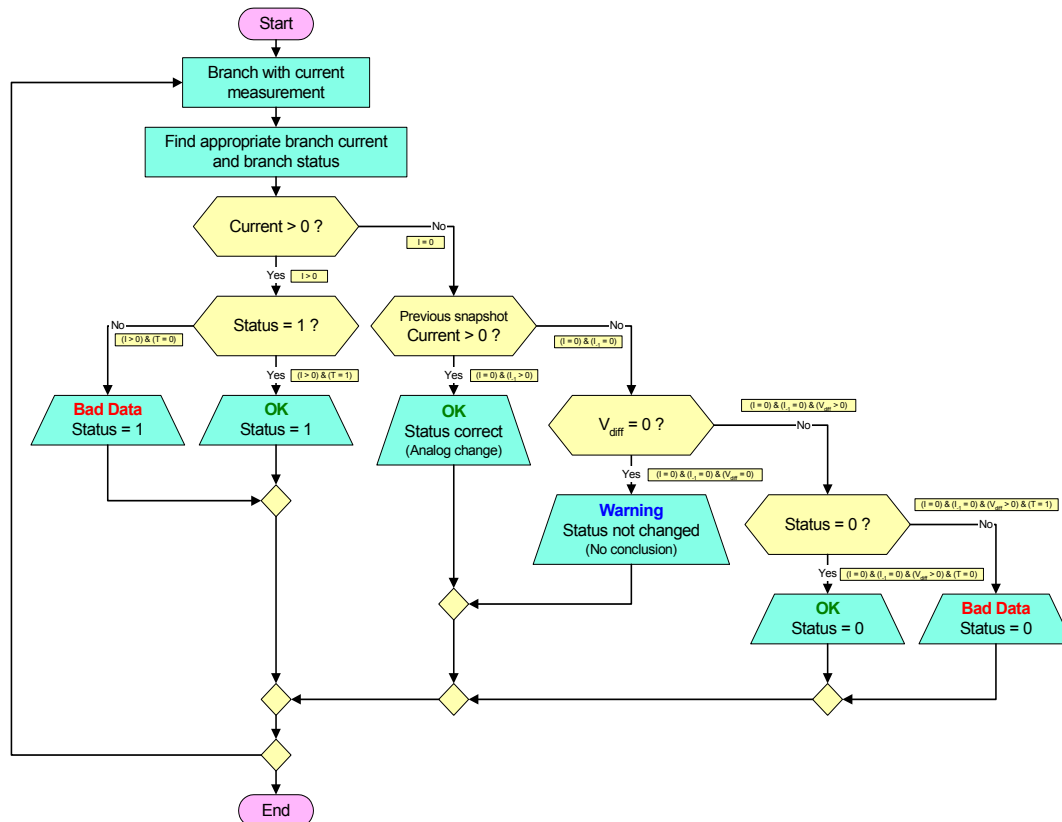


Fig. 2-11. Algorithm for branch current and status consistency check

The algorithm passes through all branches that have current measurement and retrieves the corresponding branch current value and branch status determined by the previous algorithm. The first thing that is checked is whether there is a current flow through the selected branch. In order to consider a current as a zero flow, a certain threshold is introduced through variable **ZCV** (Zero Current Value). In other words, the current needs to be less than **ZCV** to be considered as zero flow. This was necessary because there is always a small leaking current even through an open switch device.

The variable **ZCV** defines the acceptable zero value tolerance and it is initiated in the software main routine (default value of 0.0002). It can be changed at any time through the user interface.

If there is a non-zero current through the branch, the branch status is considered to be “1”. If it was “1” before this step was reached, it is not changed; i.e., it is only confirmed by the existence of the branch current. Consistency is fulfilled. In an opposite case, status needs to be corrected from “0” to “1”. The appropriate processing report is generated (e.g.

24 Bad data in Br 4 corrected to 1). This is the case when bad data is detected and eliminated. Since it is less likely that an analog measurement will become non-zero, then the status measurement will be flipped; an analog measurement is trusted more. This kind of compromise is inevitable in the case of two data sources. With more redundancy, better decisions can be made.

The case when there is no current flow through the branch is more difficult. It entails more checks to be performed before any conclusion can be reached. The first thing that is checked in this case is the value of the branch current in the previous snapshot. This is to allow temporary loss of analog measurement; i.e., the branch status will be preserved as is and it will not be influenced by erroneous analog measurement. Another assumption is that bad data will not be introduced at the moment of an analog change; one snapshot is left as a reserve before any corrective action is taken.

When the current persists at the zero level for at least two snapshots in a row, more data needs to be introduced before the branch status can be determined. At this time, the voltage difference between branch terminals is calculated. The algorithm also scans if the appropriate voltage measurements exist in the substation.

The calculated voltage difference (or the absolute value of phasor voltage difference) is compared with variable **NVD** - Necessary Voltage Difference for current existence. **NVD** is initiated in the software main routine (default value of 0.0001) and it can be changed at any time through the user interface. It determines minimum voltage that would create non-zero current through the branch. By decreasing the value of **NVD** greater sensitivity can be achieved.

If there is no voltage difference between branch terminals and since the branch current is zero, no conclusion can be made. An example of the report that is generated in this case is

5 Branch 7 --> I=0, V3-V6=0

. In this case, the status of the branch is left as is. Actually it really does not matter in this case whether the status is “0” or “1” since it would make no difference even if it is erroneous.

In cases where enough voltage difference exists and the branch current is still zero, status is determined to be “0”. If it was “0” before this step was reached, it is not changed; i.e., it is only confirmed by the existence of sufficient voltage difference while the current is zero. Therefore, the consistency is fulfilled. In cases where the status was “1”, it is corrected. Bad data are detected and eliminated by flipping the branch status. A similar processing report is generated as in the previous bad data case.

This algorithm generates an informative processing report in cases where the branch status is “1” but there is no current flow through the branch, e.g.

28 NO flow thru closed switch in Br 12

. Since this report is not even a warning, it can be ignored. It is generated only to inform the user that there is a transmission line connected to the bus which is most likely not connected to any other transmission line.

- **Time series consistency check**

This algorithm performs a consistency check change from the previous state. An analog set of measurements (both currents and voltages) and topology data are examined. Their values are compared between the last computed snapshot and the previous snapshot.

The assumption is that only a change in topology can cause a change in analog measurements. What is a change of an analog measurement should be taken conditionally. Since there are always fluctuations in power flows even in normal operation of a power network, variable *MTAMC* is introduced (default value of 0.01 pu). It determines the Maximal Tolerable Analog Measurement Change. Any change in analog measurement less than *MTAMC* is actually not considered as a change.

The flowchart of the algorithm is shown in Fig. 2-12. Changes after the previous shot need to be detected first. All branches are examined for a change in current measurements, whereas nodes are examined for change in voltage measurements. Even one change will cause the variable *AnalogChange* to memorize that there was a change in some analog measurement. Branch numbers with changed current values and node numbers with changed voltage values are stored for later reporting. Next, all branches are examined for change in topology. Even a change in the state of one switch element will cause the variable *TopologyChange* to memorize it.

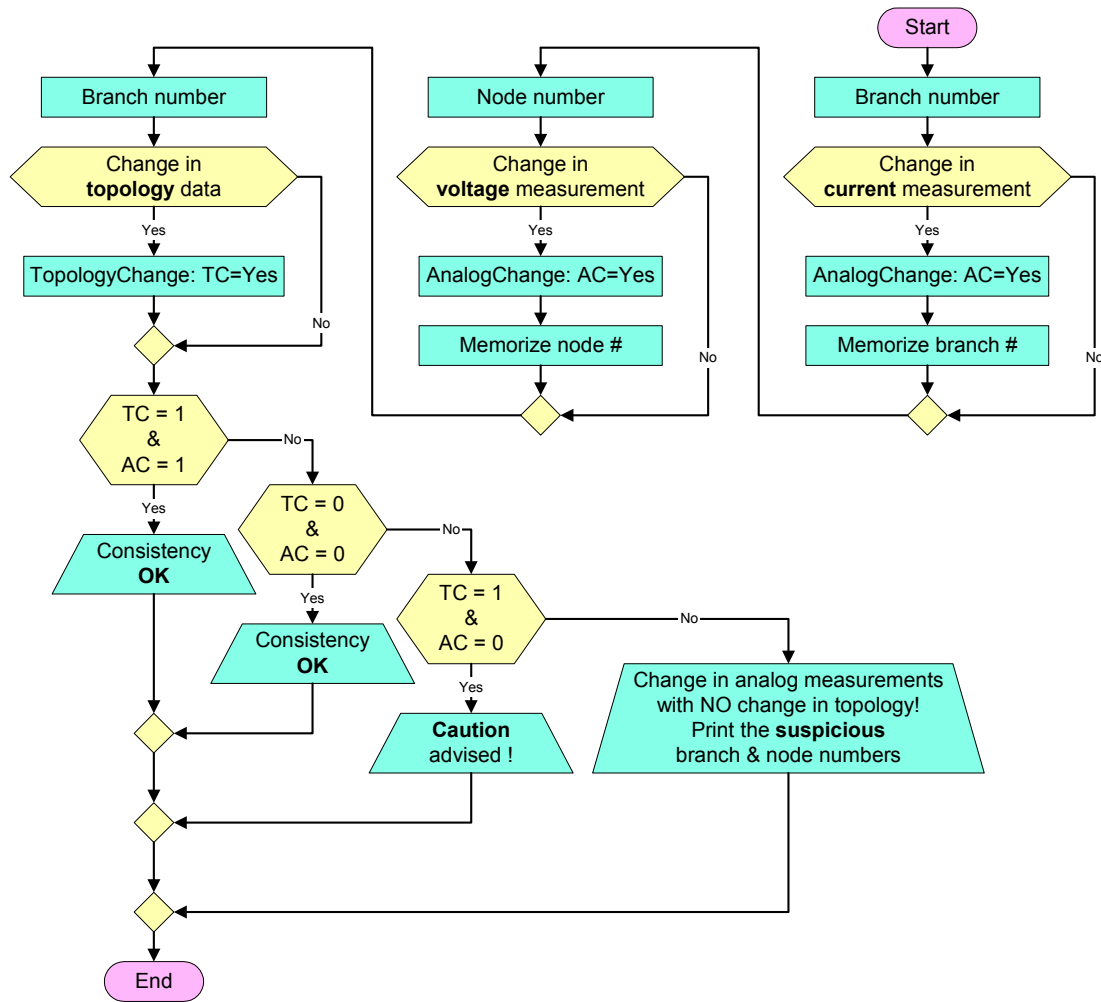


Fig. 2-12. Algorithm for consistency check based on time series

Now, the algorithm is ready to examine all four combinations of variables *AnalogChange* and *TopologyChange* for consistency. Two basic combinations yield the consistency:


- Change in topology and change in analog measurement values, and
- No change in topology and no change in analog measurements.

No action is performed in any of those two cases since consistency is fulfilled. No processing report is generated due to avoid unnecessary cluttering on the screen.

Next is the case where there is a change in topology (status) but no analog measurement has changed its value. This can happen, for example, when the branch status changes from “1” to “0” and there was no current in the branch before it was opened. Another example is a change in the status of a disconnect switch with a circuit breaker in the branch in status

“0”. Whatever the reason, this case is not considered as significant. Therefore, a single warning report is generated (blue color), e.g. **32 Status changed but NOT analogs**.

One more case exists when there is a change in some analog measurement with no change in topology. This is considered to be a more serious case than the previous one since it is more likely that the bad data is causing it. An example of the generated processing report is:



```
12 Analogs changed but NOT status
12 Check Br: 1 4 7 9 10 12
12 Check Nd: 3
```

As it can be seen, the processing report consists of a red alert line and a list of suspicious branch and node numbers. Those numbers actually designate branches with changed currents and nodes with changed voltages. It is possible that, due to some remote fault, a sudden change in power flow caused the change in analog measurements and no activity is necessary. On the other hand, rechecking the suspicious branches and nodes can discover some local instrument malfunction.

Finally, the last thing to mention is the case where all algorithms are executed with no processing report generated (i.e., neither a warning nor an alert). To confirm successful data acquisition and no error found throughout processing, the following report is generated **2 Everything OK in this snapshot**. Most of the time while running the software, these reports will be the only ones outputted; that would make other reports even more visible. The real substation processing computer could have audio signal associated with reporting to draw additional attention of substation operators.

2.3 References

- [1] SIMULINK Manual: "Using SIMULINK", The Math Works Inc., Version 4 (Release 12), November 2000
- [2] Math Works, "Using MATLAB", fourth printing, revised for MATLAB 5.3., January 1999
- [3] A. Monticelli, "State Estimation in Electric Power Systems: A Generalized Approach", *Kluwer Academic Publishers, Boston*, 1999
- [4] A. Monticelli, "Electric Power System State Estimation", *Proceedings of the IEEE*, Vol. 88, No. 2, February 2000
- [5] A. T. Johns, S. K. Salman, "Digital Protection for Power Systems", *Peter Peregrins Ltd., London, United Kingdom*, 1995
- [6] A. G. Phadke, J. S. Thorp, "Computer Relaying for Power Systems", *Research Studies Press*, 1988
- [7] P. M. Anderson, "Power System Protection", *Power Math Associates, Inc., McGraw-Hill & IEEE Press, New York*, 1999

3. State Estimation and Topology Errors

3.1 Formulation of State Estimation and Bad Data Detection

The problem of state estimation is usually formulated as a weighted least squares (WLS) problem [1], which is solved by efficient numerical techniques. The objective function to be minimized is chosen as the weighted sum of squares of the measurement residuals. However, WLS method is highly sensitive to bad data in the measurement set [2]. In order to avoid this, a different formulation of the state estimation problem has been used. It defines the sum of the weighted absolute values of the measurement residuals as the objective function. Due to its automatic bad data rejection property, LAV estimation method will be used in topology errors identification. In the following parts of this section, we will describe these two methods briefly.

3.1.1 WLS Methods

The measurement equation for a system modeled at the bus/branch level will take the following form:

$$z = h(x) + e \quad (3-1)$$

where:

z is the measurement vector of dimension m ;

$h(x)$ is the nonlinear relating the error free to the system states;

x is the state vector of dimension n ;

e is the measurement noise vector; and

n, m : the number of the state variables and measurement respectively.

Substituting the first order Taylor expansion of $h(x)$ around some x_0 in Eq.(3-1), we will have:

$$\Delta z = H \cdot \Delta x + e \quad (3-2)$$

where:

$$\Delta z = z - h(x_0)$$

$$H = \frac{\partial h}{\partial x} \text{ at } x_0$$

$$\Delta x = x - x_0$$

The weighted least square (WLS) estimate for x can be found by minimizing the following objective function:

$$J(x) = \sum_{i=1}^m \omega_i^2 (z_i - h_i(x))^2 \quad (3-3)$$

With the first order of Taylor expansion of $h(x)$ shown in (3-2), the following equation will be solved iteratively to find the solution minimizing (3-3):

$$\left. \frac{\partial J(x)}{\partial x} \right|_{x=x^k} = H' \cdot W \cdot H \cdot \Delta x^k - H' \cdot W \cdot \Delta z^k = 0 \quad (3-4)$$

where: W is the diagonal weight matrix.

Equation (3-4) can be rewritten as:

$$\Delta x^k = G^{-1} \cdot H' \cdot W \cdot \Delta z^k = 0 \quad (3-5)$$

where: $G = H' \cdot W \cdot H$ is called the gain matrix.

The WLS estimation problem given by (3-3) and (3-5) can be solved iteratively until $|\Delta x^k|$ becomes smaller than a threshold.

3.1.2 WLAV Methods

The weighted least absolute value (WLAV) estimate for x can be found by minimizing the following objective function:

$$J(x) = \sum_{i=1}^m \omega_i |z_i - h_i(x)| \quad (3-6)$$

This is accomplished by iteratively solving the following linear programming (LP) problem at each iteration k :

$$\min J(x) = \sum_{i=1}^m \omega_i (u_i + v_i) \quad (3-7)$$

$$\text{subject to} \quad \Delta z^k = H(x^k) \cdot \Delta x^k + u - v \quad (3-8)$$

where:

$$\Delta z^k = z - h(x^k)$$

$$H(x^k) = \frac{\partial h}{\partial x} \text{ at } x^k$$

ω_i : measurement weight assigned to the i th measurement; and

u, v = nonnegative slack variables such that $(u-v)$ represents the measurement residuals.

The WLAV program represented by (3-7) and (3-8) can be solved using the linear programming (LP) method.

3.1.3 Bad Data Detection/Identification

When a state estimation program fails to yield accurate estimates, it is either due to erroneous measurements, to a modeling error, or both. The former is normally known as bad data, and the latter is called the topology error.

A common technique used for bad data processing is the normalized residuals test (r^n test). In this section, we will describe the r^n test used in WLS method and WLAV method, respectively.

Normalized residuals in WLS method

Assume that the state estimate \hat{x} has already been computed from (3-5). The residuals of the measurements are defined as:

$$\hat{r} = z - h(\hat{x}) \quad (3-9)$$

The relationship between the residuals and the measurement errors can be obtained as:

$$\hat{r} = S \cdot e \quad (3-10)$$

where: $S = I - H \cdot G^{-1} \cdot H' \cdot W$.

Then the covariance matrix of the residuals can be computed by:

$$R_{\hat{r}} = S \cdot R_z \cdot S' = S \cdot R_z \quad (3-10)$$

The normalized residuals can be obtained as:

$$r^n = (\text{diag}(R_{\hat{r}}))^{-1/2} \cdot \hat{r} \quad (3-10)$$

If there exist some normalized residuals greater than the probability threshold, bad data will be detected in the measurement set. Furthermore, the measurement with the largest r^n will be identified as bad data in most of the cases.

Normalized residuals in WLAV method

The bad data processing in WLAV method follows the same procedure as in the WLS method. A detailed description of how to detect/identify bad data in WLAV method can be found in [1].

3.2 Topology Error Detection and Identification

The accuracy of state estimation highly relies on a correct model of the power system network. Any errors in the telemetered or manually updated status of circuit breaker or switch will lead to an incorrect bus level network model. Such topological errors need to be detected and identified in order to maintain a reliable database for the state estimator.

There are numerous papers addressing detection and identification of topology errors. Several rule-based methods are presented in [4,5,6]. A correlation index was proposed in [7] as an indication of possible topology errors. Other researchers [8,9] tried to identify topology errors directly based on the normalized residuals given by conventional state estimation. Some other methods are also provided in [10,11,12].

The incorporation of circuit breaker statuses within the state estimation was given by [13,14]. Several topology error identification algorithms [2,3,15,16,17,18] are proposed based on this technique. The statuses of circuit breakers are added to the state vector and estimated after the computation. This needs representation of substations at the detailed circuit breaker level. However, in order to keep the computational cost within the reasonable limits, detailed models are employed only for small sub-networks surrounding the suspected CBs. A two-stage state estimation is used for this purpose. After the first-stage state estimation, which is a conventional one, an identification procedure will be conducted to point out whether there is a topology error and where it is. The second stage will incorporate the detailed model of the suspect area and give the estimated status of the CBs. The second stage's algorithm will work well given a correctly identified area.

3.3 State Estimation Using the Substation Model

Regardless of the solution method used, the conventional state estimation formulation is based on the bus/branch model obtained from the topology processor. The circuit breakers will not appear in the model. Estimation of the power flows through circuit breakers has first been suggested for data validation at the substation by Irving and Sterling [19]. This requires the detailed topology of the substation, including the circuit breakers, to appear in the system model. Circuit breakers are modeled as zero impedance branches. Their flows are treated as additional state variables [13] because the conventional SE cannot handle the

zero impedance branches. Correspondingly, the formulation of the SE must be modified. In the following part, we will discuss how to include the substation model in the WLAV state estimation formulation.

If a substation is to be modeled in detail, representing the individual circuit breakers and their configuration, then the linearized measurement equations shown in (3-2) will take the following form:

$$\Delta z = H \cdot \Delta x + M \cdot f + e \quad (3-11)$$

where:

$[M]$ is a $(m \times l)$ measurement to circuit breaker incidence matrix defined as follows:

If the measurement i is an injection:

$$M_{ij} = \begin{cases} 1 & \text{if the injection is at the to - end of the breaker } j \\ -1 & \text{if the injection is at the from - end of the breaker } j \\ 0 & \text{otherwise} \end{cases}$$

If the measurement i is a line flow:

$$M_{ij} = \begin{cases} -1 & \text{if the flow is at the to - end of the breaker } j \\ 1 & \text{if the flow is at the from - end of the breaker } j \\ 0 & \text{otherwise} \end{cases}$$

l is the number of the circuit breakers.

f is a $(l \times 1)$ vector of power flows through the circuit breaker.

When all circuit breakers are open, then $f=0$, and Eq.(3-11) reduces to Eq. (3-2). To simplify the notation, a new vector is defined to designate the state vector augmented by the circuit breaker power flows:

$$\Delta y = [\Delta x^T f^T]^T \quad (3-12)$$

Now, the LP problem given by Eq.(3-7),(3-8) can be modified to include the breaker flow variables f , yielding:

$$\min J(x) = \sum_{i=1}^m \omega_i (u_i + v_i) \quad (3-13)$$

$$\text{subject to} \quad \Delta z^k = H(x^k) \cdot \Delta y^k + u - v \quad (3-14)$$

Additional constraints are appended to the LP problem in the form of zero voltage drops across closed circuit breakers. It is very easy to add the constraints into the measurement set in the WLAV formulation. Since the status of the breakers are not known a priori, such constraints are made soft by introducing a pair of slack variables so that the constraints will be disregarded if the breakers are actually open. For a circuit breaker between buses j and k , the following equation will be appended to Eq. (3-14):

$$x_j - x_k + u_{m+1} - v_{m+1} = 0 \quad (3-15)$$

where u_{m+1} v_{m+1} and are the nonnegative slack variables for the newly added pseudo-measurement.

Depending on the column rank of the matrix $[H|M]$, some or all entries in f will be observable. The unobservable states can be identified by the WLAV estimator during the initial phase of the solution. The details of the WLAV estimator implementation are given in [2].

3.4 Modeling of the Substation

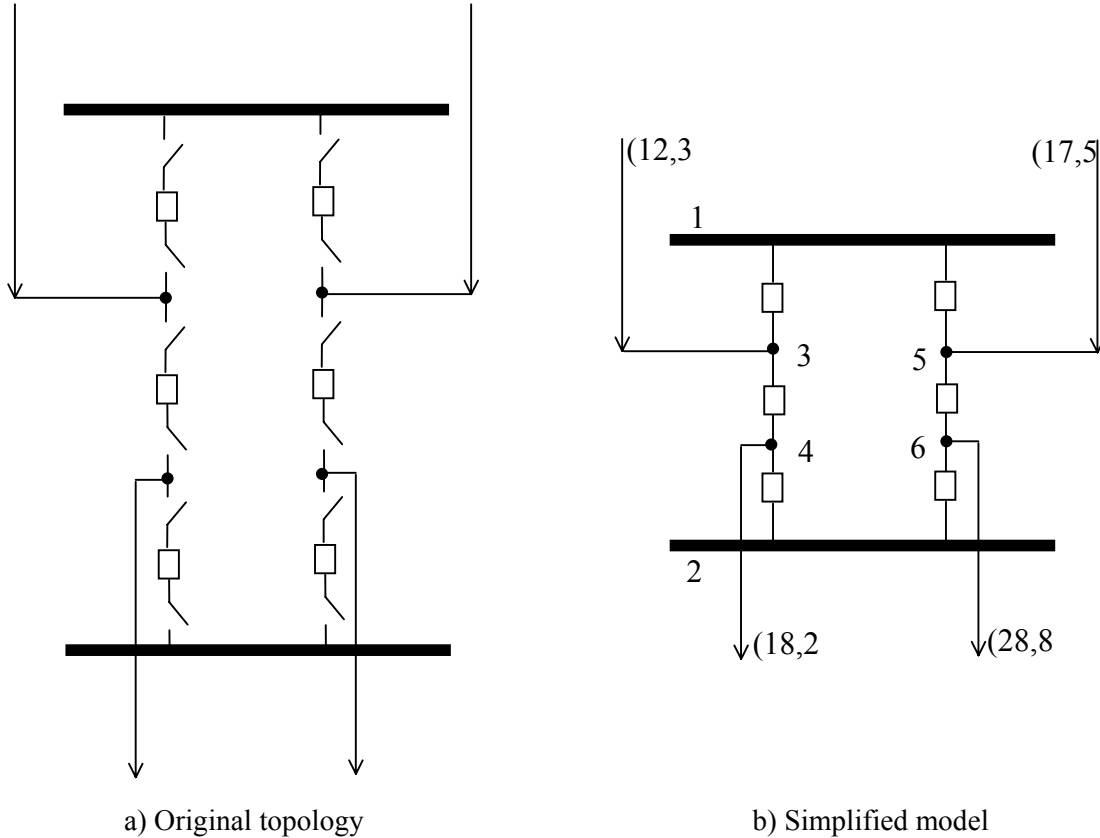


Fig. 3-1. Simplified substation model

Due to the complexity of the fully blown substation model, a simplified version that satisfies the requirements of SE will be used. The simplified model will have the following properties:

- *Every substation will be assigned a unique global number;*
- *A substation is precisely defined as that which is considered one electrical node when all breakers or switches are closed; and*
- *All the non-independent switches and circuit breakers will be considered as one zero impedance branch.*

Fig. 3-1 is an example of how to simplify a substation with a breaker and a half scheme. The detailed definition of the simplified substation model is given below.

Data files format:

The topology data and measurement data for a substation will be separated into two different files. The advantage of doing so is that we can store only one copy of topology data file for those substations which have the same configuration. The format of the topology data file and measurement data file will be shown in section A and section B, respectively.

A. Substation Topology Data File

The purpose of this data file is to describe the nodes and the connections between them. It is composed of two sections.

- **Node Data Section**

Total Number of Nodes
NodeNumber Type

NodeNumber: Node number is unique in a specified substation; and
 Type: The type of a node will be busbar (1), external (2) or internal(3).

- **Zero Impedance Branch Data Section**

Total Number of Zero Impedance Branches
FromTo

From: From node number of this branch; and
 To: To node number of this branch.

The topology data file corresponding to the substation shown in Fig. 3-1 is given below.

```

6          **Total number of nodes
1 1        ** NodeNumber      Type
2 1
3 2
4 2
5 2
6 2        ** End of node data section
6          **Total number of zero impedance branches
1 3        **From          To
1 5
2 4
2 6
3 4
5 6        ** End of Zero Impedance Branch Data Section

```

B. Measurement Data File for a Substation

This file contains the measurement data for a single substation. The first two sections are used to describe the topology information. Other sections describe the branch status and measurement data.

- **File Name of Substation Topology Data File**

Topology Data File Name

Topology Data File Name: A string that refers to the data filename containing the substation topology.

- **Inter-substation Connectivity Data**

ConSubNo ConNodeNo

.....

ConSubNo: The global number of the neighboring substation connected to this node; if none, this value will be -1; and

ConNodeNo: The node number of the neighboring substation connected to this node; if none, this value will be -1.

(Remark: The number of rows in the section and the order they are listed should be identical to the number of external nodes defined in the topology data file.)

- **Branch Status**

Status WSta

.....

Status: The status of this set of switches, 1: closed; 0:open; 2: unknown; and
WSta: Standard deviation will be assigned to the equality constraint
pseudo-measurement.

(Remark: In the same order as listed in topology data file)

- **Voltage Measurement**

Total Number of Voltage Measurements

NodeNo VMag SubWV

.....

NodeNo: Node number of this measurement;
VMag: The value of this measurement; and
SubWV: Standard deviation of this measurement.

- **Power Injection Measurement**

Total Number of Power Injection Measurements

NodeNo Pinj Qinj WInj

.....

NodeNo: Node number of this measurement;
Pinj: Measurement of active power injection of this node;
QInj: Measurement of reactive power injection of this node; and
WInj: Standard deviation of this measurement.

- **Power Flow Measurement**

Total Number of Power Flow Measurements

BranNo PFlow QFlow WFlow

.....

BranNo: Branch number of this measurement (if greater than 0, the power flow will be at the from end; if less than zero, the power flow will be at the to end);
 PFlow: Measurement of active power flows of this branch;
 QFlow: Measurement of reactive power flows of this branch; and
 WFlow: Standard deviation of this measurement;

- **Current Measurement**

Total Number of Current Measurements

BranNo BranCur WCur

.....

BranNo: Branch number of this measurement (if greater than 0, the power flow will be at the from end; if less than zero, the power flow will be at the to end);
 BranCur: Measurement of current magnitude of this branch; and
 WCur: Standard deviation of this measurement.

Consider the substation shown in Fig. 3-1 with the following information:

- Topology data file: type1.otp
- Connectivity: As shown in Fig. 3-1
- Branch Status: Closed: (1-3)(2-4)(1-5)(2-6); Opened: (3-4)(5-6)
- Voltage Measurement: At node 1,2,3,4
- Power Injection Measurement: At Node 2,5,6
- Power Flow Measurement: At branches (1-3)(3-4)
- Current Measurement: At branches (2-4)(5-6).

Then, the corresponding measurement data file will be given as:

Type1.top

12 3

18 2

17 5

28 8

1 0.001000

1 0.001000

1 0.001000

1 0.001000

```

0  0.001000
0  0.001000
4
1  1.05294  0.001000
2  0.99874  0.001000
3  1.05294  0.001000
4  0.99874  0.001000
3
2  -0.08200  -0.02500  0.001000
5  0.00000  0.00000  0.001000
6  0.00000  0.00000  0.001000
2
1  -0.09003  -0.02462  0.001000
5  0.00804  -0.00037  0.001000
2
3  -0.26052  0.001000
6  0.45768  0.001000

```

3.5 Formulation of the Two-Stage State Estimation

The two-stage state estimation procedure mentioned in section 3.2 is used in this project. Normally, a two-stage state estimation algorithm includes three parts.

3.5.1 The First-Stage State Estimator

The first-stage state estimation is nothing but a conventional one, which is based on the bus/branch model. Most of the conventional state estimation methods capable of bad data processing could be used directly. Here we use WLAV method due to its ability to exclude bad data. The WLAV method has been described briefly in section 3.1. The detailed information for this method can be found in [1].

3.5.2 Suspect Substation Identification

After the first-stage state estimation, we need to detect and identify any existing topology errors. Normally, this detection/identification procedure is based on the normalized residual analysis. In [3], the identification criterion is the number of times for each bus to which the suspect measurements of stage 1 are incident. However, experience shows that

this identification method may fail in certain cases. The identification of the suspect area constitutes the crucial step for the two-stage algorithm. Some new identification methods based on the use of normalized residual given by the first-stage state estimation are presented in section 3.6. Their performances are tested using a topology error scenario library developed and shown in section 3.6.

3.5.3 Second-Stage State Estimation and Correction of Topology Errors

The detailed models of suspect substations are incorporated into the bus/branch model in this stage. There are numerous algorithms that can handle the detailed substation model, as described in section 3.2. Most of those second-stage algorithms will work well given a correctly identified sub-area. The general WLAV method, that is shown in section 3.3 and [3], is used in our project.

The outline of the two-stage state estimation used in our project is given below:

Stage 1:

1. Run the LAV estimator using the bus level system model formed by the topology processor based on the telemetered or assumed status of the circuit breakers in the system.
2. Compute the normalized residuals by using the measurement residual covariance matrix developed in [1]. The normalized residuals are computed only for those measurements that are rejected by the LAV estimator.
3. Identify the suspect measurements with significant normalized residuals (in all simulations a threshold of 3.0 was used). If there are none, it will be decided that no topology or analog measurement errors are present. Else, go to Stage 2.

Stage2:

1. Identify the suspicious buses based on the normalized residual analysis.
2. Introduce a detailed substation model using zero impedance branches to represent the circuit breaker based on the configuration of the corresponding substation. Use all available measurements from the substation including the circuit breaker power flows, which may not have been explicitly used in Stage 1. For instance, a flow through a circuit breaker that connects two bus sections inside the same substation may get lost if the two bus sections are modeled as a single node in the bus level model.
3. Run the LAV estimator for the entire system. Repeat the normalized residual test. Flag those measurements failing the test and declare these errors as analog measurement errors. The true topology of the system will be determined according to the estimated status of the circuit breaker based on the normalized flows through them.

The flowchart of this program is shown in Fig. 3-2.

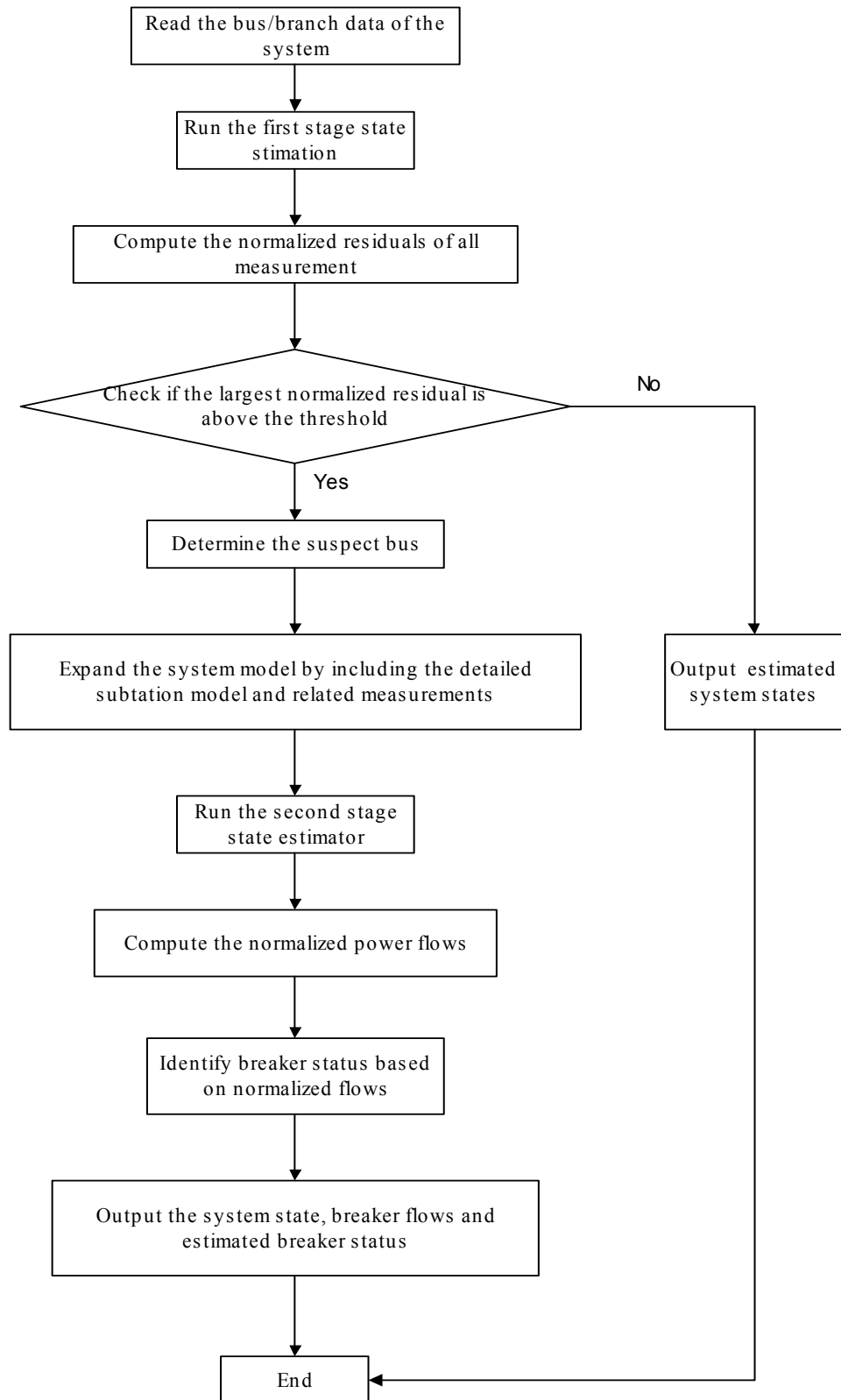


Fig. 3-2. Flowchart for the two-stage algorithm

3.6 New Suspect Substation Identification Methods

In [3], the number of times for each bus to which the suspect measurements of stage 1 are incident is computed, as shown in equation (3-16).

$$NI^i = \sum_{k=1}^{m1} I^i(z^k), i = 1 \dots n \quad (3-16)$$

where:

NI^i : Suspect measurement incidence multiplicity index for bus i ;

n : Number of buses;

$m1$: Number of suspect measurements;

z^k : k th suspect measurement; and

$I^i(z^k)$: Incident function has value get from (3-17).

$$I^i(z^k) = \begin{cases} 1 & z_k \text{ is incident to bus } i \\ 0 & \text{Otherwise} \end{cases} \quad (3-17)$$

Those buses that have the maximum NI will be identified as the suspect buses. However, during the simulation studies, it is realized that the performance of the identification procedure can be significantly improved if the directions described below are followed.

3.6.1 Considerations for Improving the Identification Performance

A. Normalized suspect measurements incidence index

Similar to (3-16) we can compute the total incidence index of the entire measurement set for each bus by (3-18):

$$NI_{Total}^i = \sum_{k=1}^m I^i(z^k), i = 1 \dots n \quad (3-18)$$

where:

NI_{Total}^i : Total incidence index of the whole measurement set for bus i ; and

m : Total number of measurements.

It is easy to show that the larger the value of NI_{Total}^i is, the more likely the corresponding bus will have a high NI^i index. As done in the case of residuals, we can normalize the suspect measurement incidence index as below:

$$NI_n^i = NI^i / NI_{Total}^i \quad (3-19)$$

where:

NI_n^i : Normalized Suspect measurement incidence index for bus i .

The performance of the suspect bus identification procedure is shown to be improved by replacing NI^i by NI_n^i .

B. Limit on the number of suspect measurements

Normally, all the measurements that have normalized residuals greater than the threshold will be treated as suspect measurements and are taken into account during the identification procedure. However, certain types of topology errors result in several measurements having large normalized residuals as well as normalized residuals close to the threshold. Such measurements do not provide useful information for the identification. Hence, only a limited number of suspect measurements should be considered, such as the top k suspect active and reactive measurements..

C. Expand the suspicious set of buses

Theoretically, if we can represent all the substations in detail (at the breaker level), we can easily identify the topology error. But we cannot afford the associated computational burden. Thus, the identification of the suspect buses is done by a two-stage procedure. However, we can increase the number of suspect buses to make sure that all those having topology errors are included. Representing several substations in detail may not significantly increase the computation burden, but may strongly increase the chances of correct identification.

In practice, the maximum number of suspect buses can be left as a parameter. For each “*maximum number*”, we can get a confidence level by a test result. The user can modify it online by taking into account the computational capability and the required confidence level.

Several alternative identification methods are designed based on the above considerations. For each of those methods, an index for each bus is computed after the first stage. The indices vector is sorted and will be used as an identification criterion. A variable, “*MaxBusNumber*”, is defined, which stands for the maximum number of suspect buses.

3.6.2 Comprehensive Identification Methods

A. Suspect only those buses with maximum index ($MaxBusNumber = 0$).

In this method, we do not specify the maximum suspect number of buses. Only those buses having the maximum indices will be identified as suspect buses. Multiple selections are allowed.

There are four possible combinations to compute the index for each bus. Correspondingly, we will have four different methods in this category.

1. *Method 0A*: Consider all the suspect measurements and use NI as the identification index. This is the method used in [3].
2. *Method 0B*: Consider only the top five suspect active and reactive measurements. Use NI as the identification index;.
3. *Method 0C*: Consider all the suspect measurements and use NI_n as the identification index.
4. *Method 0D*: Consider only the top five of suspect active and reactive measurements. Use NI_n as the identification index.

B. Suspect the buses with first n largest indices ($MaxBusNumber = n$).

In this method, we specify the maximum suspect bus number as n . Those buses having the first n largest indices will be identified as suspect buses. Multiple selections are allowed.

Similarly, we will have four different methods in this category.

1. *Method nA*: Consider all the suspect measurements and use NI as the identification index.
2. *Method nB*: Consider only the top five of suspect active and reactive measurements. Use NI as the identification index.
3. *Method nC*: Consider all the suspect measurements and use NI_n as the identification index.
4. *Method nD*: Consider only the top five of suspect active and reactive measurements. Use NI_n as the identification index.

All of these new methods are tested using the library of topology error cases described in the next section. The results will be shown in section 3.8.

3.7 Library of Topology Error Scenarios

In order to test the performance of different identification methods, a library of topology error scenarios was created based on the IEEE 30 bus system. Every electrical bus has been expanded and modeled as a hypothetical but realistic substation. All typical substation schemes have been employed in the model: ring, breaker-and-a half, etc. Fig. 3-3 shows part of the expanded test system.

For each topology error scenario, the power flow result based on the correct topology is generated. Then, the first-stage state estimation is run with the correct measurements and the wrong topology. Finally, the different identification procedures are tested.

The library contains 50 topology error cases of four types. Brief description for each type is given below.

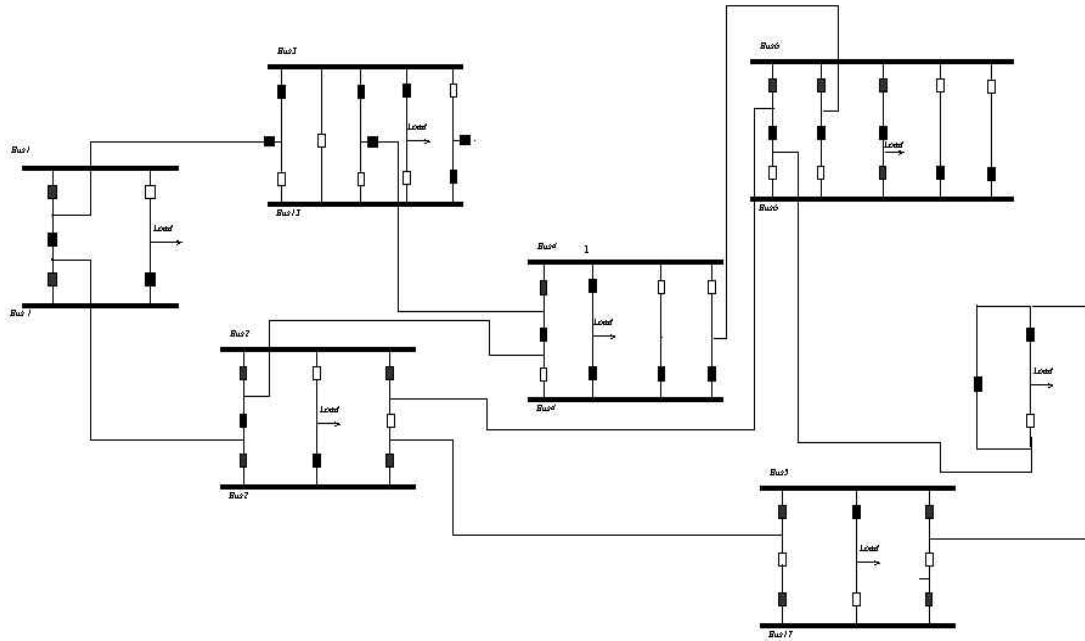


Fig. 3-3. Part of the test system

3.7.1 Type 1: Merged Bus

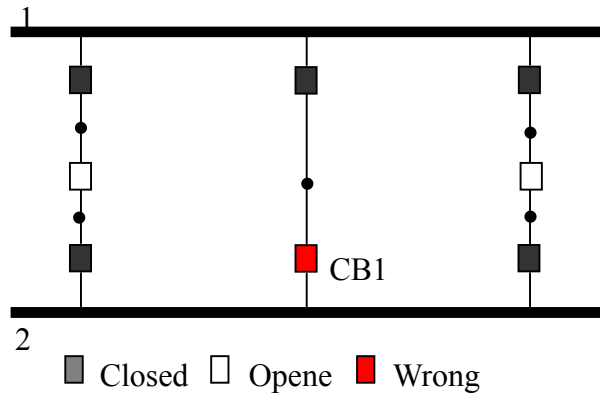


Fig. 3-4. Voltage Magnitudes Profiles

This case is shown in Fig. 3-4. The correct status of CB1 is closed. In the bus/branch model, busbar 1 and 3 should be merged as one electrical node. But the system considers the status of it as open, and split busbar 1 and 2 in the bus/branch model.

3.7.2 Type 2: Split Bus

This case is the opposite case of type 1. In Fig. 3-4, the correct status of CB1 is open. In the bus/branch model, busbars 1 and 3 should be split. But the system considers the status as closed and merges them incorrectly.

3.7.3 Type 3: Line Outage

In this case, one or both of the CBs at the ends of a transmission line is actually open, which emulates an outage of this line. The system incorrectly assumes that the line is in service.

3.7.4 Type 4: Connected Line

This case is the opposite case of type 3. The CBs at the two ends of a transmission line are actually closed. The system has the wrong status of the CB at one or both ends and incorrectly assumes that the line is disconnected.

Table 3-1 shows the number of different cases studied. Table 3-2 shows the detailed descriptions of all the cases.

TABLE 3-1
DISTRIBUTION OF THE LIBRARY

Type	1	2	3	4
Number of cases	10	10	15	15

TABLE 3-2
DESCRIPTIONS OF THE SCENARIOS

<i>CaseNO</i>	<i>Description</i>	<i>Type</i>
1	Split Bus in Sub3 (bus3,13)	2
2	Split Bus in Sub8 (bus8,29)	2
3	Split Bus in Sub9 (bus9,24)	2
4	Split Bus in Sub11 (bus11,26)	2
5	Split Bus in Sub14 (bus14,30)	2
6	Split Bus in Sub16 (bus16,19)	2
7	Split Bus in Sub18 (bus18,21)	2
8	Split Bus in Sub20 (bus20,23)	2
9	Split Bus in Sub5 (bus5,17)	2
10	Split Bus in Sub12 (bus12,31)	2
11	Merged Bus in Sub3 (bus3,13)	1
12	Merged Bus in Sub8 (bus8,29)	1
13	Merged Bus in Sub9 (bus9,24)	1
14	Merged Bus in Sub11 (bus11,26)	1
15	Merged Bus in Sub14 (bus14,30)	1
16	Merged Bus in Sub16 (bus16,19)	1
17	Merged Bus in Sub18 (bus18,21)	1
18	Merged Bus in Sub20 (bus20,23)	1
19	Merged Bus in Sub5 (bus5,17)	1
20	Merged Bus in Sub12(bus 12,31)*	1
21	Close ended line(1-3)	4
22	Close ended line(2-6)	4
23	Close ended line(4-6)	4
24	Close ended line(6-8)	4
25	Close ended line(12-13)	4
26	Close ended line(20-10)	4
27	Close ended line(14-15)	4
28	Close ended line(22-24)	4
29	Close ended line(27-30)	4
30	Close ended line(24-25)	4
31	Open ended line(1-3)	3
32	Open ended line(2-6)	3
33	Open ended line(4-6)	3
34	Open ended line(10-22)	3
35	Open ended line(12-16)	3
36	Open ended line(15-23)	3
37	Open ended line(5-7)	3
38	Open ended line(22-24)	3
39	Open ended line(27-30)	3
40	Open ended line(24-25)	3
41	Open ended line(23-24)	3
42	Open ended line(14-15)	3
43	Open ended line(21-22)	3
44	Open ended line(6-10)	3
45	Open ended line(6-9)	3
46	Close ended line(23-24)	4
47	Close ended line(21-22)	4
48	Close ended line(6-10)	4
49	Close ended line(6-9)	4
50	Close ended line(24-25)	4

*Bus 31 is a hypothetical bus, which is created for substation 12.

3.8 Simulation Results for Two-Stage State Estimation

In this section, two cases will be presented to demonstrate the two-state state estimation program's operation.

3.8.1 Case 1: Bus Split Case

This case is based on the IEEE 30 bus system (Fig. 3-5). We suppose bus 16 and bus 19, which are circled in Fig. 3-5, belong to the same substation.

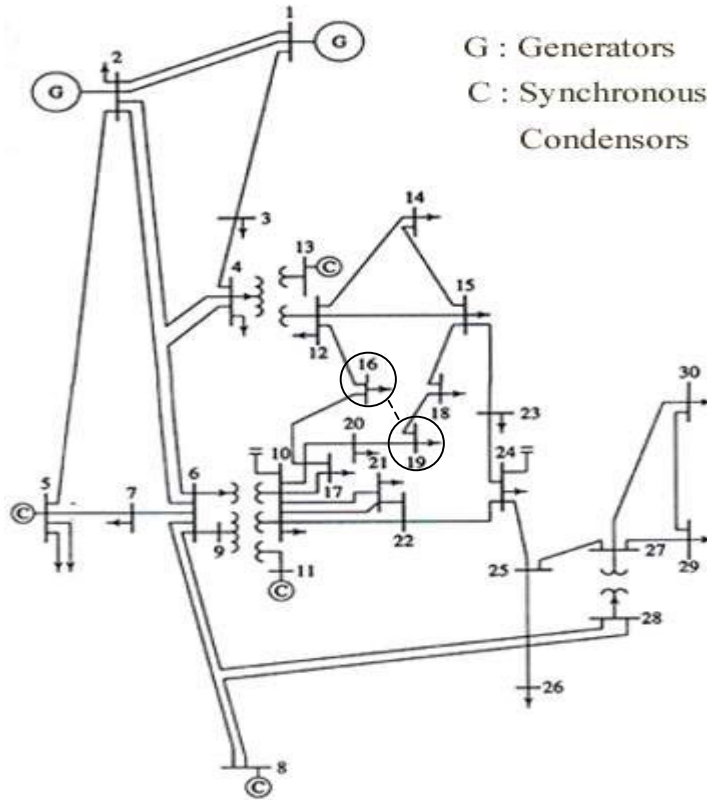


Fig. 3-5. IEEE 30 bus system

Fig. 3-6 shows the detailed topology of this substation with the status of the circuit breakers. The scenario of this demonstration is described below.

- The true status of the CB 19-33 is open, which makes this substation appears as two split buses in the bus/branch model.
- For some reasons, the operation of the CB 19-33 is not reported to the control center. The control center still assumes the status of 19-33 as closed. Then this substation will appear only as one bus in the bus/branch model.

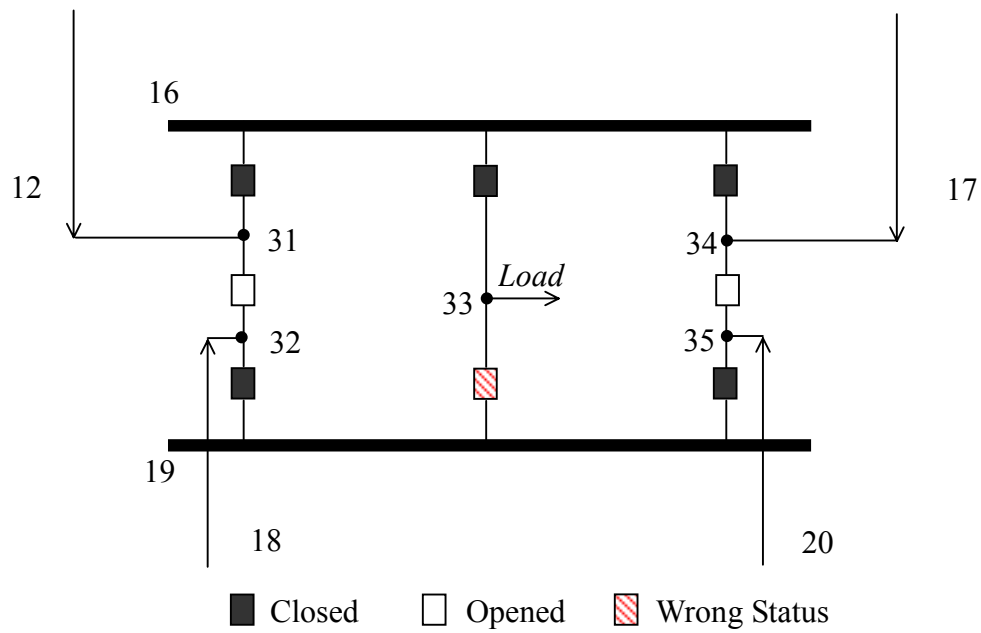


Fig. 3-6. Detailed topology for bus 16 and 30

TABLE 3-3
NORMALIZED P-RESIDUALS AFTER FIRST STAGE

<i>Meas #</i>	<i>Type</i>	<i>Location</i>	<i>Norm Res.</i>
22	FLOW	15 - 18	-2.7036
23	FLOW	18 - 16	-2.5284
25	FLOW	10 - 19	-2.0782
24	FLOW	16 - 19	2.0417
21	FLOW	16 - 17	-1.8005
26	FLOW	10 - 17	1.7402
19	FLOW	12 - 16	1.0808

TABLE 3-4
NORMALIZED Q-RESIDUALS AFTER FIRST STAGE

<i>Meas #</i>	<i>Type</i>	<i>Location</i>	<i>Norm Res.</i>
25	FLOW	10 - 19	-4.4762
24	FLOW	16 - 19	4.3675
19	FLOW	12 - 16	3.9088
21	FLOW	16 - 17	-3.8329
26	FLOW	10 - 17	3.3611
22	FLOW	15 - 18	-3.3404
23	FLOW	18 - 16	-2.9938
18	FLOW	12 - 15	-2.4542
86	VOLT	16	-1.0595
89	VOLT	19	1.0130

The first-stage state estimation uses the bus/branch model that assumes no split in bus 16 and bus 19. We can get the normalized residuals of each measurement after the first-stage state estimation. The measurements with significant residuals (more than 3.0) are listed in Table 3-3 (real power) and Table 3-4 (reactive power) in descending order.

From Tables 3-3 and 3-4 it is evident that among the buses to which the list of suspect measurements are incident, bus 16 appears the most number of times. Thus, bus 16 is identified as the most likely substation where a topology error might exist. This substation is then modeled in detail in the second stage. The substation configuration shown in Fig. 3-6 is used to expand the system model.

The second-stage state estimator is run based on the expanded system. The estimated state of this system is given in Table 3-5. The estimates for the normalized flows through the circuit breakers and their estimated status are given in Table 3-6.

TABLE 3-5
ESTIMATED STATES AFTER SECOND STAGE

<i>Bus No.</i>	True State		Estimated State	
	Magnitude	Angle	Magnitude	Angle
1	1.0600	0.00	1.0600	0.00
2	1.0430	-5.35	1.0430	-5.35
3	1.0206	-7.53	1.0206	-7.53
4	1.0116	-9.28	1.0116	-9.28
5	1.0100	-14.16	1.0100	-14.16
6	1.0103	-11.05	1.0103	-11.05
7	1.0024	-12.85	1.0024	-12.85
8	1.0100	-11.80	1.0100	-11.80
9	1.0513	-14.04	1.0513	-14.04
10	1.0458	-15.61	1.0458	-15.61
11	1.0820	-14.04	1.0820	-14.04
12	1.0563	-15.04	1.0563	-15.04
13	1.0710	-15.04	1.0710	-15.04
14	1.0427	-15.83	1.0427	-15.83
15	1.0401	-15.84	1.0401	-15.84
16	1.0355	-16.03	1.0355	-16.03
17	1.0373	-15.95	1.0373	-15.95
18	1.0376	-16.04	1.0376	-16.04
19	1.0391	-15.97	1.0391	-15.97
20	1.0400	-15.93	1.0400	-15.93
21	1.0335	-16.05	1.0335	-16.05
22	1.0340	-16.04	1.0340	-16.04
23	1.0291	-16.24	1.0291	-16.24
24	1.0227	-16.42	1.0227	-16.42
25	1.0180	-16.01	1.0180	-16.01
26	1.0003	-16.43	1.0003	-16.43
27	1.0236	-15.49	1.0236	-15.49
28	1.0069	-11.67	1.0069	-11.67
29	1.0038	-16.72	1.0038	-16.72
30	0.9923	-17.61	0.9923	-17.61
31			1.0355	-16.03
32			1.0391	-15.97
33			1.0391	-16.03
34			1.0355	-16.03
35			1.0391	-15.97

TABLE 3-6
ESTIMATED STATE OF CIRCUIT BREAKER

<i>Circuit Breaker</i>	<i>Normalized P Flow</i>	<i>Normalized Q Flow</i>	<i>Estimated Status</i>
16-31	-115.2171	-49.0266	Closed
32-32	0.1701	-0.0076	Open
19-32	12.7813	5.6847	Closed
16-33	125.4597	55.9197	Closed
19-33	0.0987	0.0987	<u>Open</u>
16-34	-4.6246	-6.7945	Closed
34-35	-0.0568	0.0137	Open
19-35	-12.7813	-5.6847	Closed

These estimated states reported in Table 3-6 are consistent with the simulated topology error, correctly identifying the bus split that has taken place. The underlined estimated status means that there is a topology error identified after the second stage.

3.8.2 Case 2: Line Outage Case

This case is based on the IEEE 30 bus system (Fig. 3-7). We suppose bus 16 and bus 19 belong to the same substation. There will be a topology error in the line 12-16, which is dotted in Fig. 3-7.

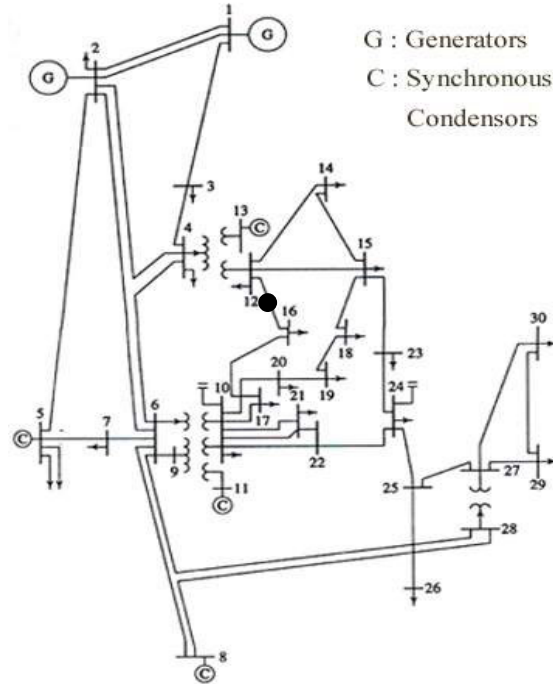


Fig. 3-7. IEEE 30 bus system

Fig.3-8 shows the detail topology of this substation. The scenario of this demo is described below.

- The true status of the CB 16-31 is open, which mean the transmission line 12-16 is outage in the bus/branch model.
- For some reason, the operation of the CB 16-31 is not reported to the control center. The control center still assumes the status of 16-31 is closed. The transmission line 12-16 is still assumed to be in use in the system model.

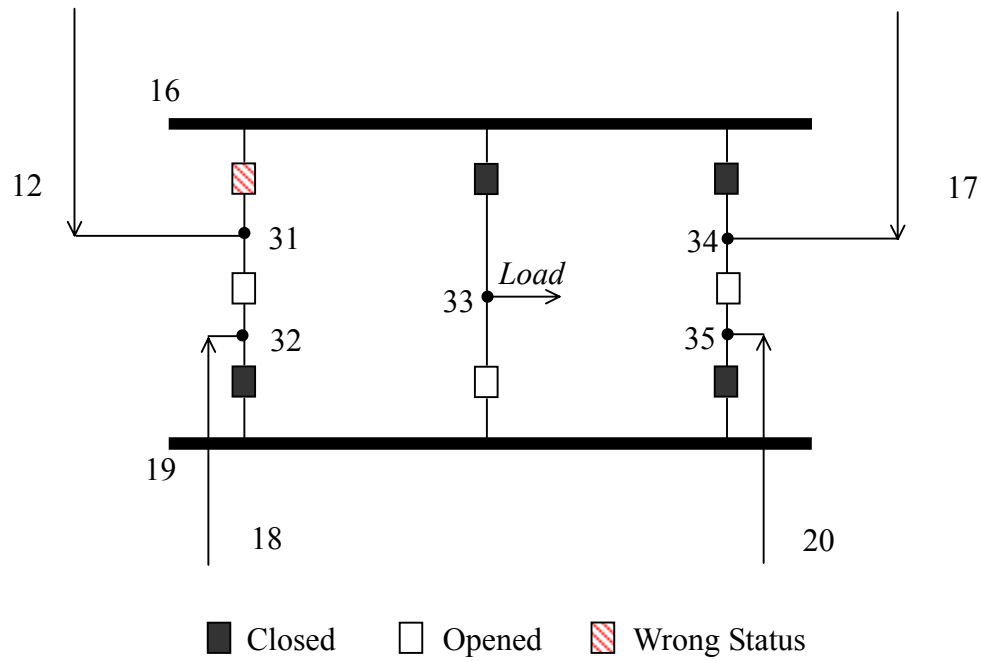


Fig. 3-8. Detailed topology for bus 16 and 19

The first-stage state estimation uses the bus/branch model that assumes transmission line 12-16 is still on. We can get the normalized residuals of each measurement after the first-stage state estimation. The measurements that have significant residuals (more than 3.0) are listed in Table 3-7 (real power) and Table 3-8 (reactive power) in descending order.

TABLE 3-7
NORMALIZED P-RESIDUALS AFTER FIRST STAGE

<i>Meas #</i>	<i>Type</i>	<i>Location</i>	<i>Norm Res.</i>
19	FLOW	12 - 16	-63.9720
21	FLOW	16 - 17	-62.5336
53	INJ	12	-27.8090
58	INJ	17	23.0791

TABLE 3-8
NORMALIZED Q-RESIDUALS AFTER FIRST STAGE

<i>Meas #</i>	<i>Type</i>	<i>Location</i>	<i>Norm Res.</i>
87	VOLT	16	-13.0589
19	FLOW	12 - 16	-11.7973
21	FLOW	16 - 17	-10.3564
53	INJ	12	-6.3620
58	INJ	17	2.7513

From Tables 3-7 and 3-8 it is evident that among the buses to which the list of suspect measurements are incident, bus 16 appears the most number of times. Thus, bus 16 is identified as the most likely substation where a topology error might exist. This substation is then modeled in detail in the second stage. The substation configuration shown in Fig. 3-8 is used to expand the system model.

The second-stage state estimator is run based on the expanded system. The estimated state of this system is given in Table 3-9. The estimates for the normalized flows through the circuit breakers and their estimated status are given in Table 3-10.

The estimated states reported in Table 3-10 are consistent with the simulated topology error, correctly identifying the bus split that has taken place. The underlined estimated status means that there is a topology error identified after the second stage.

TABLE 3-9
ESTIMATED STATES AFTER SECOND STAGE

<i>Bus No.</i>	True State		Estimated State	
	Magnitude	Angle	Magnitude	Angle
1	1.0600	0.00	1.0599	0.00
2	1.0430	-5.37	1.0429	-5.37
3	1.0211	-7.52	1.0209	-7.52
4	1.0122	-9.26	1.0120	-9.27
5	1.0100	-14.22	1.0099	-14.22
6	1.0095	-11.13	1.0094	-11.13
7	1.0019	-12.93	1.0018	-12.93
8	1.0100	-11.90	1.0099	-11.90
9	1.0461	-14.52	1.0460	-14.52
10	1.0359	-16.31	1.0358	-16.31
11	1.0820	-14.52	1.0819	-14.52
12	1.0612	-14.28	1.0611	-14.28
13	1.0710	-14.28	1.0709	-14.28
14	1.0464	-15.24	1.0463	-15.25
15	1.0413	-15.41	1.0412	-15.41
16	1.0006	-18.35	1.0005	-14.28
17	1.0188	-17.11	1.0187	-17.11
18	1.0347	-16.01	1.0346	-16.01
19	1.0339	-16.17	1.0338	-16.17
20	1.0335	-16.26	1.0334	-16.26
21	1.0242	-16.68	1.0241	-16.68
22	1.0250	-16.65	1.0249	-16.65
23	1.0275	-16.11	1.0274	-16.11
24	1.0176	-16.72	1.0175	-16.72
25	1.0147	-16.25	1.0146	-16.25
26	0.9969	-16.67	0.9969	-16.67
27	1.0214	-15.70	1.0213	-15.70
28	1.0060	-11.77	1.0059	-11.77
29	1.0015	-16.93	1.0015	-16.93
30	0.9900	-17.82	0.9901	-17.82
31			1.0611	-14.28
32			1.0339	-16.17
33			1.0005	-14.28
34			1.0005	-18.35
35			1.0338	-16.17

TABLE 3-10
ESTIMATED STATE OF CIRCUIT BREAKER

<i>Circuit Breaker</i>	<i>Normalized P Flow</i>	<i>Normalized Q Flow</i>	<i>Estimated Status</i>
16-31	0.0963	0.0963	<u>Open</u>
32-32	-0.0556	0.0565	Open
19-32	-20.7539	4.3846	Closed
16-33	129.8345	57.8696	Closed
19-33	-0.0577	0.1444	Open
16-34	-129.8386	-57.8714	Closed
34-35	0.0589	0.0570	Open
19-35	20.7550	-4.3849	Closed

3.9 Simulation Results for Suspect Substation Identification Methods

The library is used to test for methods with $MaxBusNumber = 0$ (*Method 0A-0D*) and $MaxBusNumber = 3$ (*Method 3A-3D*). The results are showed in Table 3-11 and Table 3-12, respectively.

Column 2 in the result tables lists the bus numbers that should be identified as suspect buses. Columns 4-7 list the suspicious buses obtained from the different methods. If at least one bus in column 2 is included in the result bus list, we would say that it is correct. Otherwise, it is not correct, and will be underlined and italicized. The final rows of the tables are the statistical results of each method. The numerator represents the number of cases that can be identified correctly by this method while the denominator represents the total number of cases.

From the results, we can see all of the methods can correctly identify suspicious bus in cases of *type 2* and *type 3*. For these two types, status of CBs is assumed to open while it is actually closed. However, most of the methods do not show good performance for the counterpart situation without increasing the number of suspicious buses.

It is easy to see from the result that the best performance was with the method that utilized the normalized suspect measurement incident number. On the other hand, only taking into account the top five suspect active and reactive measurements will also improve the performance.

By increasing the number of suspicious buses slightly, we can get significant improvement without significantly increasing the computation burden. *Method 3D* has only three suspicious buses and correctly identifies suspicious buses for all 50 cases.

TABLE 3-11
TEST RESULT FOR METHOD 0X

<i>Case NO</i>	<i>Correct Bus</i>	<i>Type</i>	<i>Suspected bus</i>			
			<i>Method 0A</i>	<i>Method 0B</i>	<i>Method 0C</i>	<i>Method 0D</i>
1	3;13	2	<u>12</u>	<u>4</u>	<u>12;20</u>	3;4
2	8;29	2	<u>27</u>	<u>27</u>	8	<u>27</u>
3	9;24	2	<u>10</u>	<u>22</u>	22;24;28	<u>22</u>
4	11;26	2	<u>25;28</u>	<u>25</u>	<u>25;28</u>	9;11;25;26
5	14;30	2	<u>27</u>	<u>27</u>	<u>27</u>	<u>27</u>
6	16;19	2	19	16	19	16
7	18;21	2	<u>10</u>	<u>10;15</u>	<u>20</u>	21
8	20;23	2	<u>10</u>	23	<u>18</u>	23
9	5;17	2	<u>6</u>	5;7;10	<u>2</u>	5;7
10	12;31	2	<u>14;15</u>	<u>14;15</u>	<u>14</u>	<u>14</u>
11	3;13	1	3;13	3;13	13	13
12	8;29	1	8;29	8;29	8;29	8;29
13	9;24	1	9;24	9;24	9	9
14	11;26	1	11;26	11;26	11;26	11;26
15	14;30	1	14;30	14;30	14;30	14;30
16	16;19	1	16	16	16	16
17	18;21	1	21	21	21	21
18	20;23	1	20;23	20;23	20	20
19	5;17	1	5;17	5;17	5;17	5;17
20	12;31	1	12	12	12	12
21	1;3	4	1;3	1;3	3	3
22	2;6	4	2;6	2;6	2	2
23	4;6	4	4;6	4;6	4	4
24	6;8	4	6;8	6;8	8	8
25	12;13	4	12;13	12;13	13	13
26	20;10	4	20	20	20	20
27	14;15	4	14;15	14;15	14	14
28	22;24	4	22;24	22;24	22;24	22;24
29	27;30	4	27;30	27;30	30	30
30	24;25	4	24;25	24;25	24;25	24;25
31	1;3	3	<u>2</u>	<u>2</u>	<u>2</u>	<u>2</u>
32	2;6	3	2	2	<u>5;7</u>	<u>5</u>
33	4;6	3	4	<u>2</u>	4	<u>2;3</u>
34	10;22	3	<u>21;23;24</u>	<u>21;23</u>	<u>21</u>	<u>21</u>
35	12;16	3	<u>10</u>	<u>17</u>	<u>4</u>	<u>17</u>
36	15;23	3	23;24	23	23	23
37	5;7	3	<u>2</u>	<u>6</u>	<u>2</u>	2;7
38	22;24	3	24	<u>15;23;25;27</u>	24	<u>23</u>
39	27;30	3	<u>29</u>	<u>29</u>	<u>29</u>	<u>29</u>
40	24;25	3	24;25;27;28	24;25;27;28	24;25;28	24;25;28
41	23;24	3	<u>15</u>	<u>15</u>	<u>15</u>	<u>15</u>
42	14;15	3	15	15	14;15	14
43	21;22	3	10;22	<u>10</u>	21;22	22
44	6;10	3	6;10	6;10	6;10	6;10
45	6;9	3	<u>10</u>	<u>10</u>	<u>10</u>	9;16;17
46	23;24	4	24	24	24	24
47	21;22	4	21;22	21;22	21	21
48	6;10	4	10	10	10	10
49	6;9	4	6;9	6;9	9	9
50	24;25	4	24;25	24;25	24;25	24;25
Total			34/50	33/50	35/50	38/50

TABLE 3-12
TEST RESULT FOR METHOD 3X

<i>Case NO</i>	<i>Correct Bus</i>	<i>Type</i>	<i>Suspected bus</i>			
			<i>Method 3A</i>	<i>Method 3B</i>	<i>Method 3C</i>	<i>Method 3D</i>
1	3;13	2	<u>12;6;10</u>	4;12;3	<u>12;20;1</u>	3;4;13
2	8;29	2	27;8;28	27;28;29	8;27;28	27;29;28
3	9;24	2	<u>10;6;22</u>	<u>22;10;21</u>	22;24;28	22;21;24
4	11;26	2	<u>25;28;6</u>	<u>25;9;27</u>	<u>25;28;8</u>	9;11;25
5	14;30	2	27;14;24	27;29;30	27;14;29	27;29;30
6	16;19	2	19;16;17	16;17;18	19;16;17	16;17;18
7	18;21	2	<u>10;15;20</u>	10;15;21	20;21;22	21;15;22
8	20;23	2	<u>10;22;24</u>	23;15;18	<u>18;22;24</u>	23;18;15
9	5;17	2	<u>6;2;10</u>	5;7;10	<u>2;28;1</u>	5;7;9
10	12;31	2	14;15;31	14;15;31	14;31;15	14;31;15
11	3;13	1	3;13	3;13	13;3	13;3
12	8;29	1	8;29	8;29	8;29	8;29
13	9;24	1	9;24	9;24	9;24	9;24
14	11;26	1	11;26	11;26	11;26	11;26
15	14;30	1	14;30	14;30	14;30	14;30
16	16;19	1	16	16	16	16
17	18;21	1	21;18	21;18	21;18	21;18
18	20;23	1	20;23	20;23	20;23	20;23
19	5;17	1	5;17	5;17	5;17	5;17
20	12;31	1	12	12	12	12
21	1;3	4	1;3	1;3	3;1	3;1
22	2;6	4	2;6	2;6	2;6	2;6
23	4;6	4	4;6	4;6	4;6	4;6
24	6;8	4	6;8	6;8	8;6	8;6
25	12;13	4	12;13	12;13	13;12	13;12
26	20;10	4	20	20	20	20
27	14;15	4	14;15	14;15	14;15	14;15
28	22;24	4	22;24	22;24	22;24	22;24
29	27;30	4	27;30	27;30	30;27	30;27
30	24;25	4	24;25	24;25	24;25	24;25
31	1;3	3	2;1;3	2;1;3	2;1;3	2;1;3
32	2;6	3	2;6;5	2;5;6	5;7;2	5;2;7
33	4;6	3	4;2;6	2;4;3	4;2;3	2;3;4
34	10;22	3	<u>21;23;24</u>	21;23;22	<u>21;23;24</u>	21;23;22
35	12;16	3	10;4;12	17;10;16	<u>4;17;10</u>	17;16;20
36	15;23	3	23;24;15	23;15;24	23;24;15	23;24;15
37	5;7	3	<u>2;4;6</u>	<u>6;2;4</u>	2;4;7	2;7;6
38	22;24	3	24;23;22	<u>15;23;25</u>	24;23;22	23;25;15
39	27;30	3	29;27;30	29;27;30	29;30;27	29;30;27
40	24;25	3	24;25;27	24;25;27	24;25;28	24;25;28
41	23;24	3	15;23;12	15;23;12	15;23;14	15;23;12
42	14;15	3	15;12;14	15;14;12	14;15;12	14;15;12
43	21;22	3	10;22;21	10;22;21	21;22;24	22;21;10
44	6;10	3	6;10	6;10	6;10	6;10
45	6;9	3	10;6;12	10;6;9	10;6;12	9;16;17
46	23;24	4	24;23	24;23	24;23	24;23
47	21;22	4	21;22	21;22	21;22	21;22
48	6;10	4	10;6	10;6	10;6	10;6
49	6;9	4	6;9	6;9	9;6	9;6
50	24;25	4	24;25	24;25	24;25	24;25
Total			<i>42/50</i>	<i>46/50</i>	<i>44/50</i>	<i>50/50</i>

3.10 References

- [1] A. Abur, "A Bad Data Identification Method for Linear Programming State Estimation", IEEE Transactions on Power Systems, Vol.5, No.3, August 1990.
- [2] A. Abur, M.K. Celik, "Topology Error Identification by Least Absolute Value State Estimation", Electrotechnical Conference, 1994 Proceedings, 7th Mediterranean, 1994.
- [3] A. Abur, H. Kim, M. Celik, "Identifying the Unknown Circuit Breaker Statuses in Power Networks", IEEE Transactions on Power Systems, Vol. 10, No. 4, November 1995.
- [4] N. Singh, H. Glavitsch, "Detection and Identification of Topological Errors in Online Power System Analysis", IEEE Transactions on Power Systems, Vol. 6, No. 1, February 1991.
- [5] N. Singh, F. Oesch, "Practical Experience with Rule-based On-line Topology Error Detection", IEEE Transactions on Power Systems, Vol. 9, No. 2, May 1994.
- [6] T. Tian, M. Zhu, B. Zhang, "An artificial neural network-based expert system for network topological error identification," in Proc. IEEE Int. Conf. Neural Networks, vol.2, pp.882-886, Nov, 1995, Australia, Perth.
- [7] A.S. Costa, J.A. Leao, "Identification of Topology Errors in Power System State Estimation", IEEE Transactions on Power Systems, Vol. 8, No. 4, November 1993.
- [8] K.A. Clements, P.W. Davis, "Detection and Identification of Topology Errors in Electric Power Systems", IEEE Transactions on Power Systems, Vol. 3, No. 4, November 1988.
- [9] F.F. Wu, W.H.E. Liu, "Detection of Topology Errors by State Estimation", IEEE Transactions on Power Systems, Vol. 4, No. 1, February 1989.
- [10] C.N. Lu, J.H. Teng, B.S. Chang, "Power System Network Topology Error Detection", IEE Proceedings - Generation, Transmission and Distribution, Vol. 141, No. 6, November 1994.
- [11] L. Mili, G. Steeno, F. Dobraca, D. French, "A Robust Estimation Method for Topology Error Identification", IEEE Transactions on Power Systems, Vol. 14, No. 4, November 1999.
- [12] Mai Hoa Vuong, S. Lefebvre, Xuan Dai Do, "Detection and identification of topological errors from real-time measurement reconciliation," in Proc. IEEE PES Winter Meeting, Jan. 26-31, 2002, New York.
- [13] A. Monticelli, "Modeling Zero Impedance Branches in Power System State Estimation", IEEE Transactions on Power Systems, Vol. 6, No. 4, November 1991.
- [14] A. Monticelli, "The Impact of Modeling Short Circuit Branches in State Estimation", IEEE Transactions on Power Systems, Vol. 8, No. 1, February 1993.
- [15] K.A. Clements, A.S. Costa, "Topology Error Identification Using Normalized Lagrange Multipliers", IEEE Transactions on Power Systems, Vol. 13, No. 2, May 1998.

- [16] A.S. Costa, F. Viera, "Topology error identification through orthogonal estimation methods and hypothesis testing [power system]," in Power tech Proceedings, vol.3, Sep. 10-13, 2001, Porto.
- [17] J.C. Pereira, J.T. Saraiva, V. Miranda, A.S Costa, E.M. Lourenço, K.A. Clements, "Comparison of approaches to identify topology errors in the scope of state estimation studies," in Power tech Proceedings, vol.3, Sep. 10-13, 2001, Porto.
- [18] Antonio Gomez Exposito, Antonio de la Villa Jaen. "Reduced Substation Models for Generalized State Estimation", IEEE Trans On Power Systems, Vol.16, No.4, November, 2001.
- [19] M. R. Irving and M. J. Sterling, "Substation data validation", IEE Proceedings, Vol.129, No.3, pp.119-122, May 1982.

4. Three Phase State Estimation-Sensitivity Study

4.1 Shortcoming of the Single Phase State Estimation

Most of the state estimation (SE) programs that used in today's modern control center are based on the following assumptions:

- 1) The system is symmetrical between phases; and
- 2) The system operates under fully balanced operating conditions.

In practice, most high voltage systems are nearly balanced and, depending on the system configuration and loading conditions, they can be modeled and solved in the positive sequence. However, there may be cases where the balanced system assumptions no longer hold, when bus loads have an uneven distribution among the three phases, or relatively long but non-transposed transmission lines carrying significant power flows exist in the system. Such lines will have different mutual coupling among the pairs of phase conductors and consequently the power flows through each of the three conductors of the lines will not be the same.

Any phase unbalances in loads and/or any existing non-transposed transmission lines are commonly ignored by the state estimators that are used for power systems today. However, such simplifying assumptions may affect the accuracy and numerical robustness of the estimator. We will study the effects of such simplifying assumptions on the estimated state of the systems under varying operating conditions.

4.2 Possible State Estimation Methods

Three possible ways of estimating the state of a system will be investigated.

4.2.1 Estimate 1 (Three-phase)

A full three-phase state estimation will be performed using the full set of three-phase measurements, assuming that the necessary instrumentation is available to have access to the measurements in all three phases. Once the three-phase estimates are obtained, their positive sequence components will be evaluated and recorded. No assumptions will be made regarding the structure and operating condition of the power system.

4.2.2 Estimate 2 (Single-phase)

In this case, it will be assumed that only phase A measurements are available at the control center where the state estimator is run. The positive sequence network model, similar to the one used by the common single-phase state estimators, will be employed.

The resulting state estimate, which is a single-phase result, will be recorded. This case is based on the assumptions mentioned above.

4.2.3 Estimate 3 (Positive sequence)

This is identical to the Estimate 2, except for the measurement set, which will now contain the positive sequence values of the three-phase measurements. This corresponds to a case where three-phase instrumentation and the corresponding measurements are available, yet a single-phase state estimator is to be run. This case requires that assumption 1) be true or almost true.

4.3 Three-Phase State Estimation Formulation and Modeling

A state estimator based on the full three-phase network model is developed. The weighted least squares (WLS) algorithm is used in the implementation of the three-phase state estimator. The details of the measurement equations and Jacobian entries can be found in [1]. Sparse matrix techniques are used to improve the computational efficiency and memory savings. All system components, such as transmission lines, loads, transformers and generators, are modeled in three-phase as described below.

4.3.1 Three Phase Transmission Lines

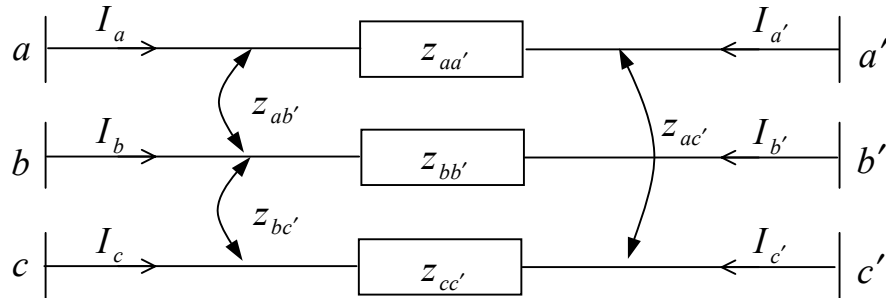


Fig. 4-1. An example of three-phase transmission line

A typical three-phase transmission line is given in Fig. 4-1. The network equations for this line can be written in compact form, according to the procedure described in [2]. The effect of the ground wire is included in the self and mutual impedance of the three-phase conductors. The primitive series impedance matrix of the line is given:

$$Z_P = \begin{bmatrix} Z_{aa'} & Z_{ab'} & Z_{ac'} \\ Z_{ba'} & Z_{bb'} & Z_{bc'} \\ Z_{ca'} & Z_{cb'} & Z_{cc'} \end{bmatrix} \quad (4-1)$$

Defining the primitive admittance matrix $Y_p = Z_p^{-1}$, the nodal equations for the system of Fig. 4-1 can be written as:

$$\begin{bmatrix} I_a \\ I_b \\ I_c \\ I_{a'} \\ I_{b'} \\ I_{c'} \end{bmatrix} = \begin{bmatrix} & & & & & \\ & Y_p & & & -Y_p & \\ & & & & & \\ & & & & & \\ -Y_p & & & & Y_p & \\ & & & & & \end{bmatrix} \begin{bmatrix} V_a \\ V_b \\ V_c \\ V_{a'} \\ V_{b'} \\ V_{c'} \end{bmatrix} \quad (4-2)$$

If the susceptances associated with line charging exist, they will be added to the diagonal elements of the admittance matrix corresponding to the end nodes. The susceptances of all shunt elements in three phases are assumed to be equal.

4.3.2 Three-Phase Loads and Generators

Each three-phase bus consists of three, single-phase buses with loads connected in Wye and modeled by negative power injections in the state estimation measurement equations. Similarly, generated real and reactive power at each single-phase bus is modeled as a positive injection. Generator buses may have unbalanced injections assigned to them as measurements if the operating conditions are not balanced.

4.3.3 Transformers

The winding-connection type of transformer becomes critically important in the three-phase study [3]. While all transformers considered in this study are assumed to be Wye connected at both sides, any other combination can be modeled as shown in [3]. Transformers with off-nominal tap settings are represented as shown in Fig. 4-2.

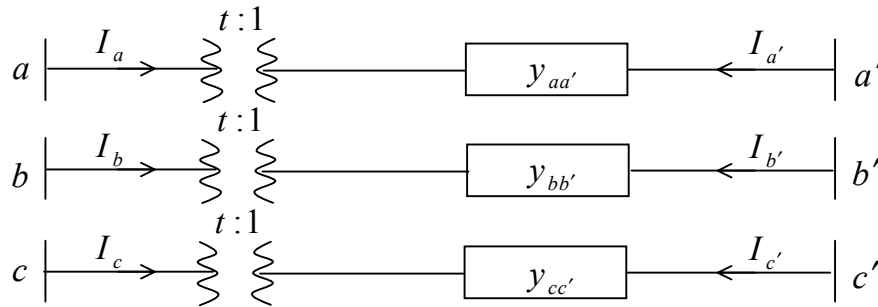


Fig. 4-2. Typical three-phase transformer model

Accordingly, the node equations of transformer can be described as (4-3).

$$\begin{bmatrix} I_P \\ I_S \end{bmatrix} = \begin{bmatrix} A/t^2 & C/t \\ C^T/t & B \end{bmatrix} \begin{bmatrix} V_P \\ V_S \end{bmatrix} \quad (4-3)$$

where:

I_P and I_S are primary and secondary three-phase current;

V_P and V_S are primary and secondary three-phase voltages; and

t is the off-nominal tap.

$$A = B = -C = \begin{bmatrix} y_{aa'} & 0 & 0 \\ 0 & y_{bb'} & 0 \\ 0 & 0 & y_{cc'} \end{bmatrix}.$$

For the admittance matrices corresponding to other kinds of winding connections, please refer to [3].

4.3.4 Bus Shunts

Bus shunts are assumed to be decoupled in each phase and they are modeled by adding appropriate susceptance values to the diagonal elements corresponding to the buses.

4.4 Sensitivity Study: Effect of Neglecting 3-Phase Unbalances

4.4.1 Study Cases

Several cases with different kinds and degree of unbalances were studied to investigate the influence of neglecting unbalances. All of these cases are based on IEEE 30 bus system. The detailed information for how to get the simulation data can be found in [4]. Following is a brief description of those cases.

1) Case T1

In this case, all transmission lines are assumed to be non-transposed. The amount of coupling asymmetry among the three-phase conductors is chosen based on the mutual impedance between phase A and phase C. This quantity is set equal to 90% of the mutual impedances between the other two phases.

2) Case T2

This case is identical to Case 1, except for the severity of the coupling asymmetry. The mutual impedance between phase A and phase C is set equal to 60% of other two mutual impedances.

3) Cases L1-L4

These are a set of four cases where all bus loads in the system are assumed to be unbalanced. The amount of unbalance between phase loads is varied by keeping the phase A and B loads equal, and changing phase C load to 90%, 80%, 70% and 60% of that of the other phase loads for the cases L1 through L4 respectively.

4.4.2 Investigation Methodology

The investigation concerning the described cases above is carried out by performing three state estimation solutions using different assumptions and available measurements, which are outlined in section 4.1.

These three sets of estimated results will be referred respectively as *Estimate1*, *Estimate2* and *Estimate3* in the presented tables below. The estimated states in *Estimate2* and *Estimate3* are compared with *Estimate1* to quantify the effects of the assumptions involved. A flowchart of overall investigation procedure is summarized in Fig. 4-3.

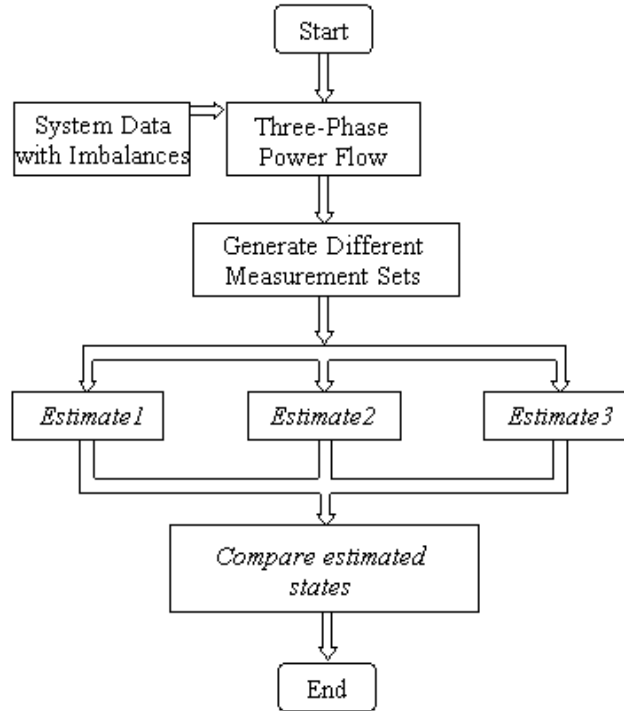


Fig.4-3. Flowchart of investigation process

4.4.3 Sensitivity Indices

The following indices are used for the comparisons:

1) *Normalized Residuals*

The normalized residuals for all measurements are computed for both kinds of state estimators. Those measurements with normalized residual great than 3.0 are recorded as the suspected bad data.

2) *Maximum Absolute State Mismatches*

Absolute mismatches between the states obtained as Estimate1 and as Estimate2, Estimate3 are computed. Maximum absolute mismatches of voltage magnitude and phase angles are recorded.

3) *Freq. of Relative Errors Greater than 3σ*

The relative errors given by (4-4) are computed for *Estimate2* and *Estimate3*.

$$err = \left| \frac{S_{estimated} - S_{true}}{S_{true}} \right| \quad (4-4)$$

where $S_{estimated}$ is the estimated states and S_{true} is true states.

The number of times these relative errors exceed three times the corresponding measurement standard deviation will be referred to as Freq. of $err > 3\sigma$.

4) *Costs*

The values of the objective functions evaluated after convergence are referred to as costs. For comparison purposes, the costs obtained from the three-phase state estimator are divided by three. The value of the cost is compared against the corresponding Chi-square test threshold in order to detect bad data. Chi-square test thresholds from a Chi-square distribution table are 139 for the single phase and 127 (383/3) for the three-phase estimation cases. Those values greater than the threshold are noted in the tables.

4.5 Simulation Results

All of the cases mentioned above are simulated and compared according to the procedure shown in Fig. 4-3. The detailed results of the estimation will be presented for *case T1* (Table 4-1), and, for brevity, only the corresponding indices calculated for the other cases will be shown. In these tables, all the voltage magnitudes will be given in per-unit and phase angles in degrees.

4.5.1 Non-Transposed Cases

In the first two cases, cases T1 and T2, the loads are balanced but the transmission lines are non-transposed. Table 4-1 shows all three estimation results for case T1. The indices of these two cases are presented in Table 4-2 and 4-3, respectively.

TABLE 4-1
ESTIMATED STATES OF CASE T1

<i>Bus No.</i>	<i>Estimate1</i>		<i>Estimate2</i>		<i>Estimate3</i>	
	<i> V </i>	<i>Ang.</i>	<i> V </i>	<i>Ang.</i>	<i> V </i>	<i>Ang.</i>
1	1.060	0.00	1.054	0.00	1.060	0.00
2	1.043	-5.40	1.039	-5.45	1.043	-5.40
3	1.020	-7.58	1.017	-7.67	1.020	-7.58
4	1.011	-9.33	1.009	-9.44	1.011	-9.33
5	1.009	-14.27	1.008	-14.39	1.009	-14.27
6	1.009	-11.11	1.007	-11.25	1.009	-11.11
7	1.001	-12.92	0.999	-13.08	1.001	-12.92
8	1.009	-11.87	1.008	-12.01	1.009	-11.87
9	1.049	-14.09	1.048	-14.22	1.049	-14.09
10	1.043	-15.64	1.041	-15.81	1.043	-15.64
11	1.080	-14.07	1.081	-14.26	1.080	-14.08
12	1.054	-14.91	1.054	-15.09	1.054	-14.90
13	1.068	-14.94	1.068	-15.16	1.068	-14.94
14	1.039	-15.82	1.038	-15.91	1.039	-15.82
15	1.035	-15.84	1.035	-16.03	1.035	-15.84
16	1.041	-15.46	1.040	-15.59	1.041	-15.46
17	1.037	-15.79	1.036	-15.94	1.037	-15.79
18	1.025	-16.45	1.025	-16.68	1.025	-16.45
19	1.023	-16.57	1.022	-16.79	1.023	-16.57
20	1.027	-16.37	1.026	-16.60	1.027	-16.37
21	1.030	-16.07	1.029	-16.24	1.030	-16.07
22	1.031	-16.06	1.030	-16.23	1.031	-16.06
23	1.025	-16.26	1.025	-16.40	1.025	-16.26
24	1.019	-16.43	1.018	-16.57	1.019	-16.43
25	1.015	-16.08	1.014	-16.09	1.015	-16.08
26	0.997	-16.35	0.999	-16.53	0.997	-16.35
27	1.020	-15.61	1.019	-15.58	1.020	-15.61
28	1.005	-11.73	1.004	-11.87	1.005	-11.73
29	0.999	-16.88	0.998	-16.73	0.999	-16.88
30	0.987	-17.75	0.985	-17.61	0.987	-17.75

TABLE 4-2
COMPARISON INDICES OF CASE T1

<i>CaseT1</i>	Bad Data No.	Maximum Mismatch		Freq. of err >3σ		Cost
		<i> V </i>	<i>Ang.</i>	<i> V </i>	<i>Ang.</i>	
Estimate1	0	-	-	-	-	101.3
Estimate2	0	0.005	0.238	0	9	117.0
Estimate3	0	4E-5	0.002	0	0	49.3

TABLE 4-3
COMPARISON INDICES OF CASE T2

<i>CaseT2</i>	Bad Data No.	Maximum Mismatch		Freq. of err $>3\sigma$		Cost
		V	Ang.	V	Ang.	
Estimate1	0	-	-	-	-	97.9
Estimate2	0	0.018	0.393	1	28	123.5
Estimate3	0	3E-4	0.031	0	0	41.5

The use of single phase (phase A) measurements to estimate the state of a system operating under unbalanced conditions may lead to some bias in the state estimate. However, for both cases T1 and T2, Chi-square test thresholds are not hit; i.e., modeling errors due to the non-transposed lines are not detected by the estimator.

4.5.2 Unbalanced Cases

TABLE 4-4
COMPARISON INDICES OF CASE L1

<i>CaseL1</i>	Bad Data No.	Maximum Mismatch		Freq. of err $>3\sigma$		Cost
		V	Ang.	V	Ang.	
Estimate1	0	-	-	-	-	90.6
Estimate2	0	0.011	0.455	0	6	125.9
Estimate3	0	4E-5	5E-4	0	0	42.5

Cases L1-L4 assume fully transposed transmission lines but unbalanced loads. Table 4-4 shows the results for case L1. The results appear similar to the ones reported for case T2, which corresponds to a very extreme form of asymmetry as compared to a relatively mild unbalance (10%) of case L1. When the degree of unbalances increases, the errors will increase significantly. Table 4-5 shows the results for cases L1-L4, where the cost function for case L4 exceeds the bad data detection threshold based on the Chi-square test. Hence, in this situation the state estimator may incorrectly identify some measurements as bad and discard them to further deteriorate the accuracy of the estimator.

TABLE 4-5
INFLUENCE OF UNBALANCED LOADS

<i>Cases</i>	Bad Data No.	Maximum Mismatch		Freq. of err $>3\sigma$		Cost
		V	Ang.	V	Ang.	
L1	0	0.011	0.455	0	6	125.9
L2	1	0.017	0.868	12	29	127.5
L3	0	0.025	1.103	19	29	131.1
L4	4	0.030	1.243	24	29	159.1*

*: This value exceeds Chi-square test threshold.

Fig. 4-4 shows how the error increases with the severity of unbalances.

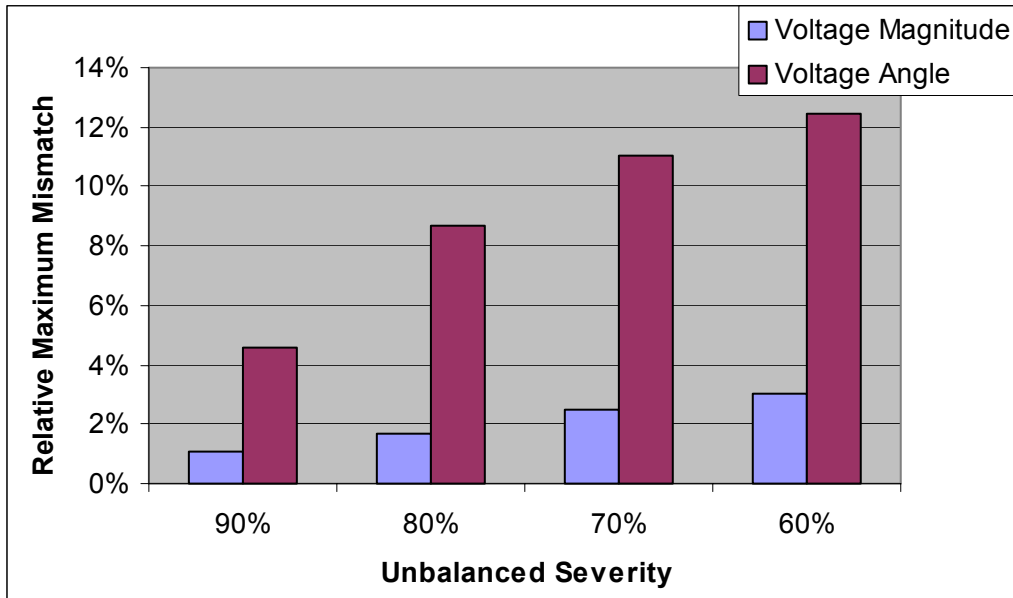


Fig. 4-4. Relative error caused by unbalances with different degrees

It is evident from these simulation results that the effect of non-transposed lines is less than that of load unbalances on the state estimates. Fig. 4-5 and 4-6 show the effects of non-transposed lines (caseT2) and unbalanced loads (case L4) on the voltage magnitude and phase angle estimates respectively. In both figures, the curve labeled *estimate1* is the true state while the one labeled *estimate2* represents the estimate. The mismatch between these two curves reflects the error caused by the assumptions.

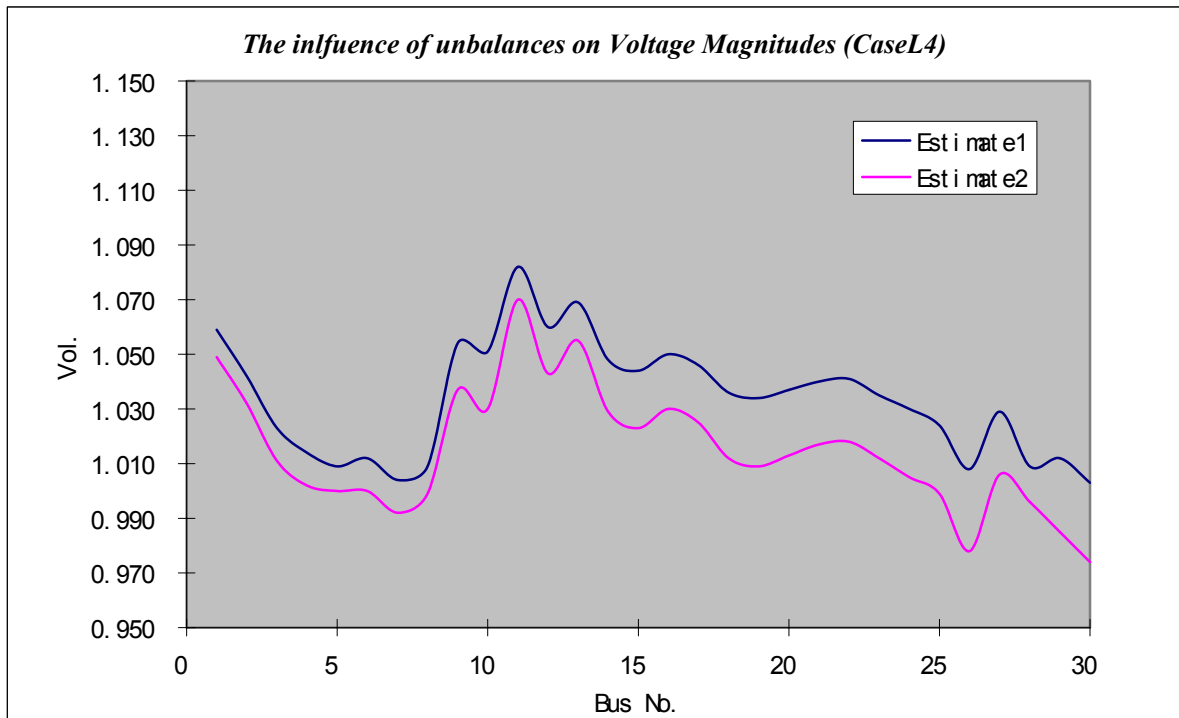
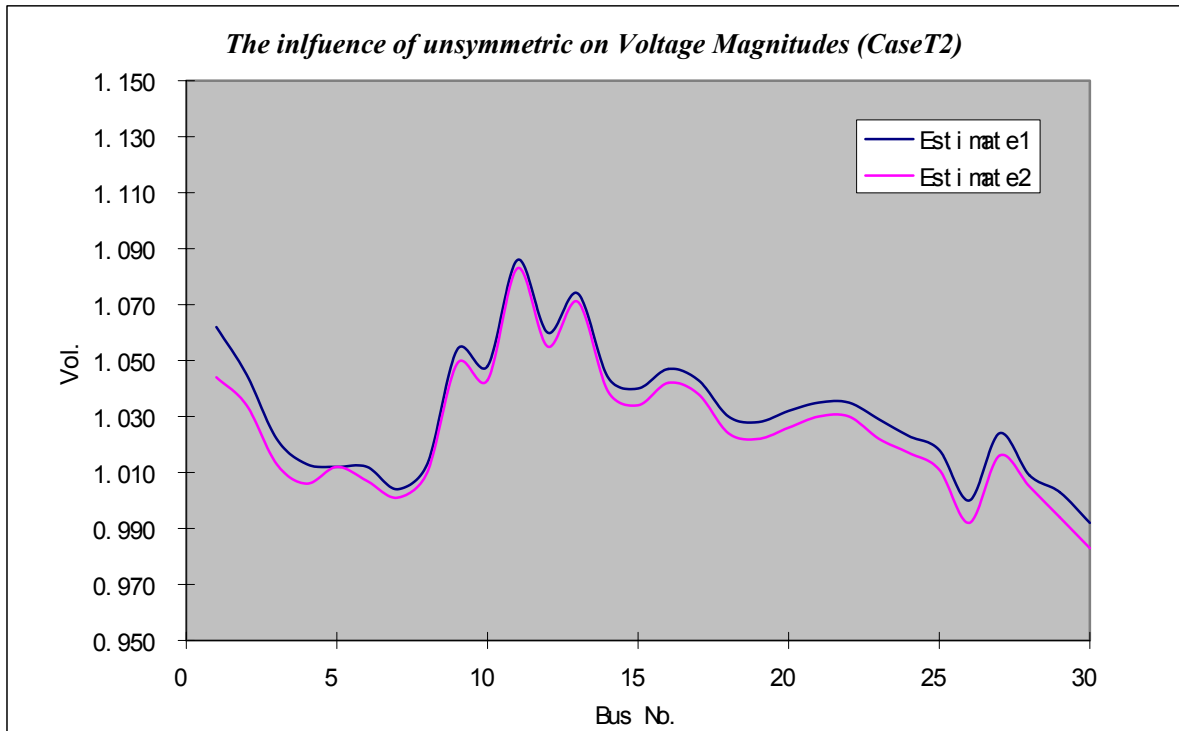


Fig 4-5. Influences of unsymmetric and unbalances on voltage magnitudes

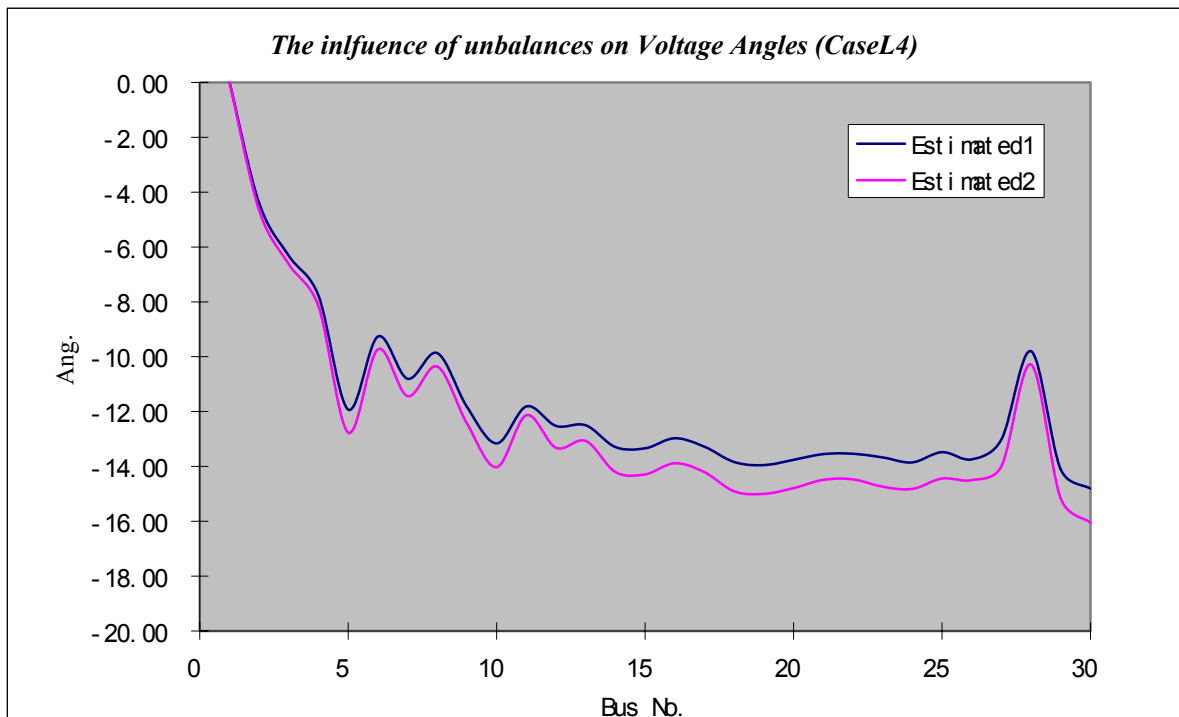
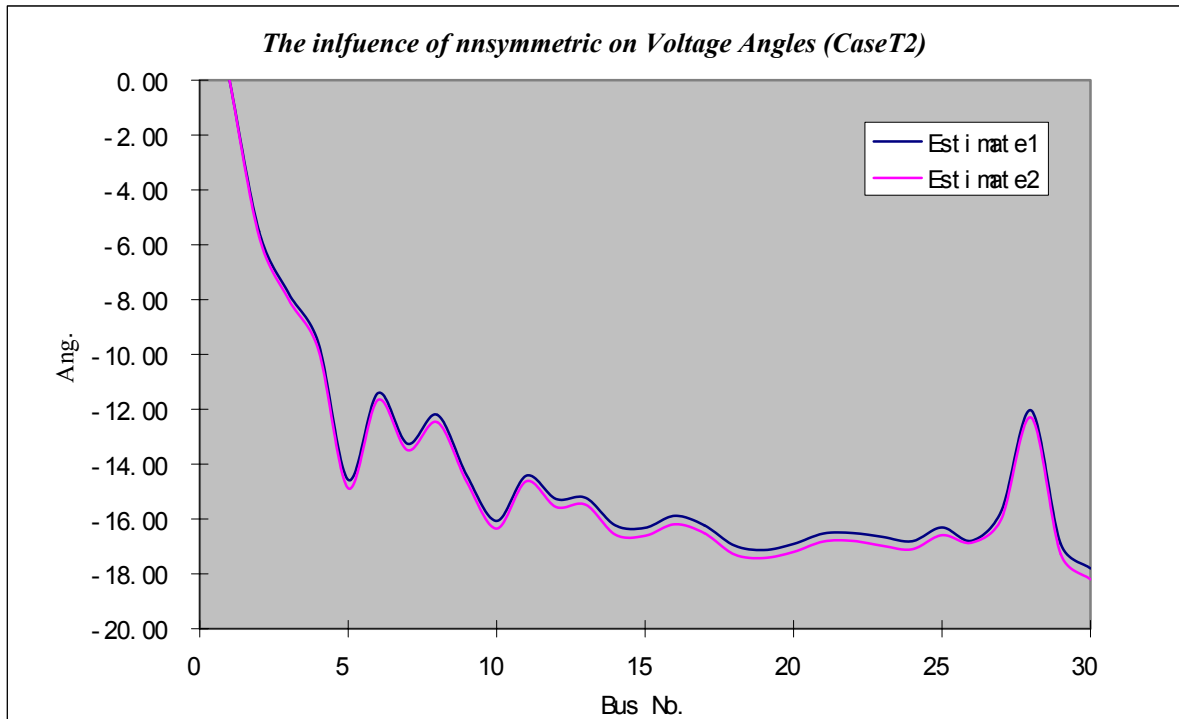


Fig .4-6. Influences of unsymmetric and unbalances on voltage angles

4.5.3 Conclusion of the Sensitivity Studies

The results of the sensitivity studies will be illustrated in the test results section of this report. It is demonstrated that under certain cases, the use of single-phase state estimator may lead to significant biases in the solution due to existing asymmetries or load unbalances. The simulation results also indicate a higher sensitivity of the system state to loading unbalances than to asymmetries in the transmission line conductor configurations.

4.6 References

- [1] H. Kim and A. Abur, "State estimation for three phase power networks", Proceedings of the 26th Annual North American Power Symposium, Sep. 27-28, 1994, Manhattan, Kansas, pp. 210-220.
- [2] M. Chen and W. E. Dillon, "Power system modeling," Proceedings of the IEEE, vol. 62, pp. 902-915, Jul. 1974.
- [3] C. W. Hansen and A. S. Debs, "Power system state estimation using three-phase models," IEEE Trans. Power System, vol. 10, pp. 818-824, May. 1995.
- [4] S. Zhong and A. Abur, "Effects of non-transposed lines and unbalanced loads on state estimation," in Proc. 2002 IEEE PES Winter Meeting, Jan. 26-31, 2002, New York.

5. Graphical Interface for the Two-Stage State Estimation

A program called Power Education Toolbox (PET.) was upgraded to support the two-stage state estimation function. PET is a user friendly software package that was developed at Texas A&M University as a teaching tool for power engineering. The software is designed to provide easy access to several commonly used application functions (such as power flow analysis, state estimation, etc.) using the same user interface and power system diagrams. A Windows-based graphical user interface program provides the link between the user and the various application programs. Fig. 5-1 shows the main windows of PET. For detailed functions and usage of PET, please refer to “Power Education Toolbox (PET) Users Manual”[1].

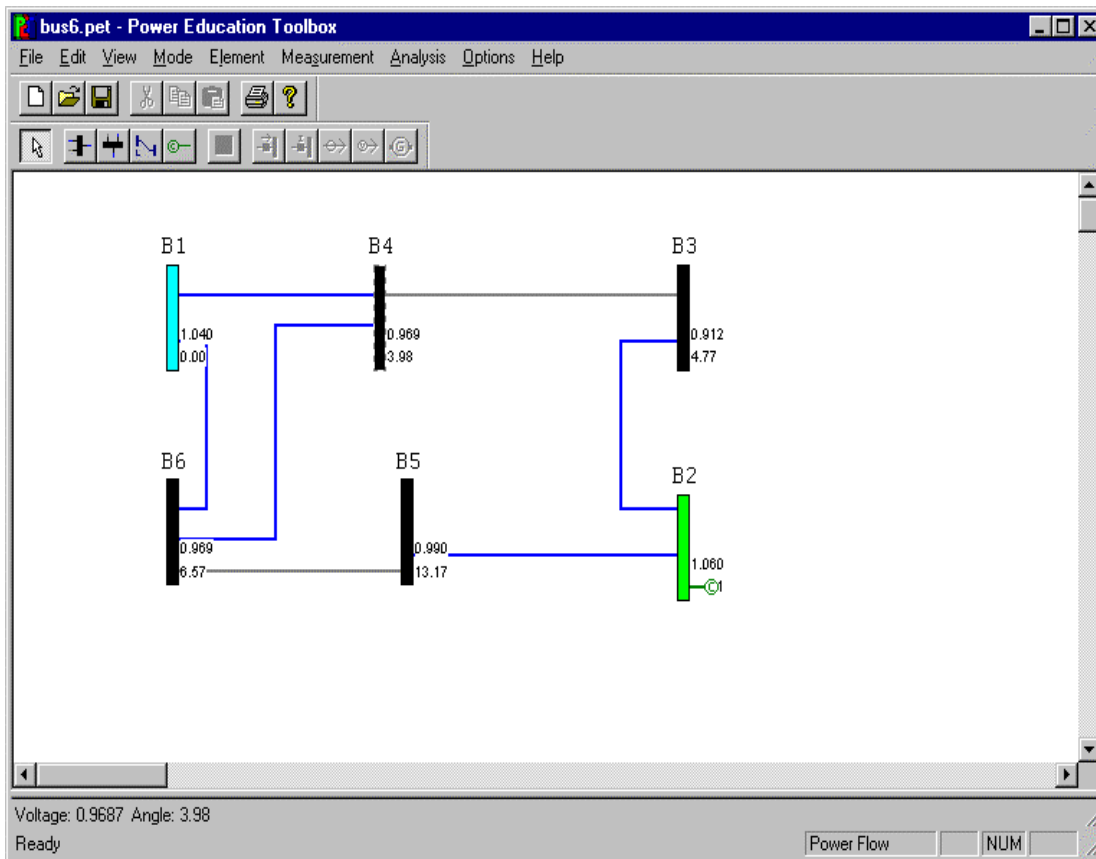


Fig. 5-1. Main windows of PET

Current version of PET only provides the conventional state estimation function. In order to support the topology error identification function, an application program that implements the general WLAV method shown on section 3 was developed. The graphical user interface was also modified correspondingly.

5.1 Auxiliary Substation Window

For the purpose of topology error identification, we needed the detailed substation models for suspect substations. An auxiliary substation window for editing and setting the substation model was created. Fig. 5-2 shows the diagram of PET with the auxiliary window open.

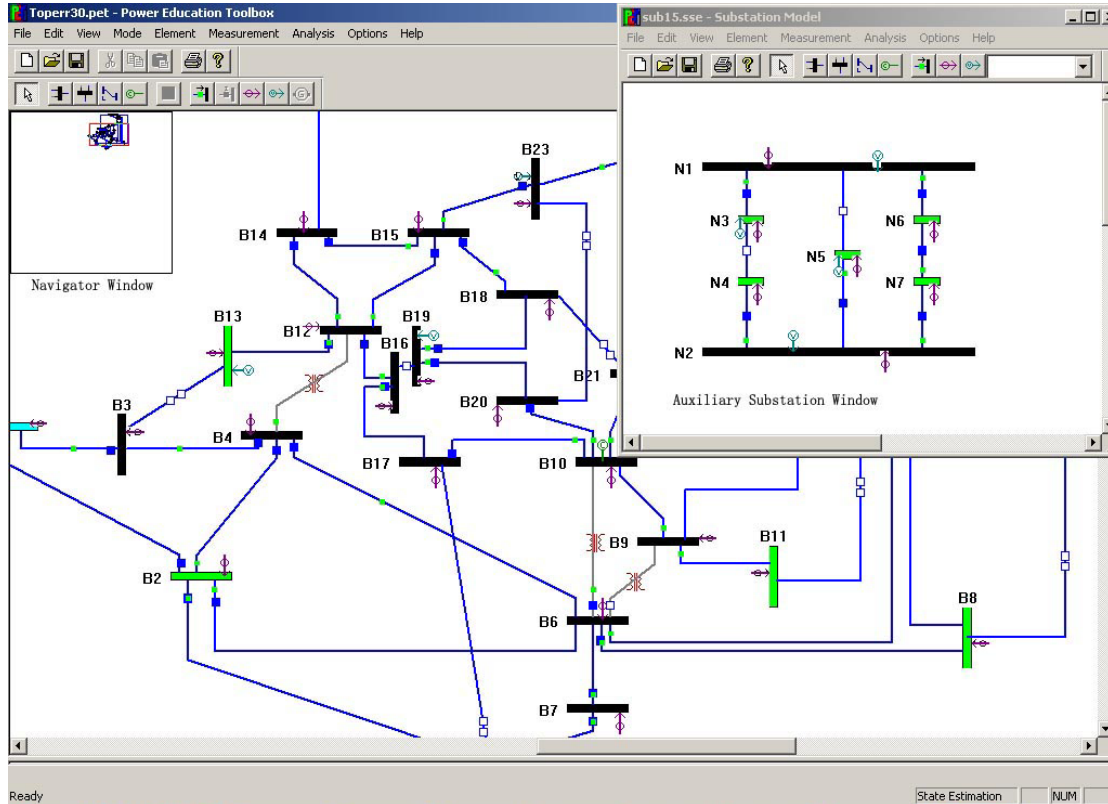


Fig. 5-2. PET with the auxiliary Substation window

The link between the buses in the main window and the substation model can be set through the modified property dialog boxes in the main window and the new property dialog boxes of the auxiliary window. The following part of this section will introduce the modified interface elements and the procedure for detailed parameter setting.

5.2 Editing the Substation Model

5.2.1 Creation of the Substation Model from Scratch

The substation model support by PET is defined in section 3.4. The auxiliary window is hidden by default when PET is executed. There are two ways to open the auxiliary window for editing.

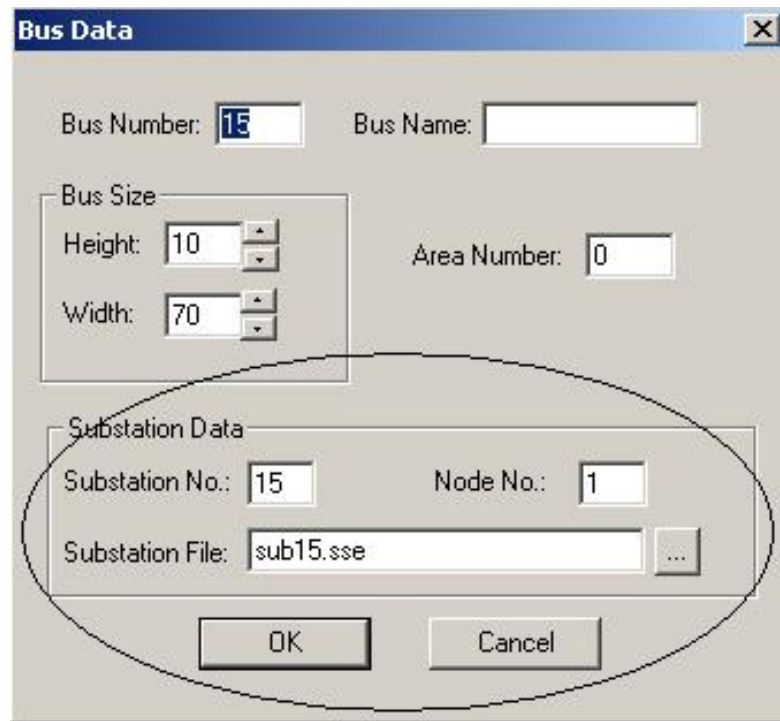


Fig. 5-3. Bus's Property Dialog Box with Substation Data

1. Open the bus's property dialog box. For example, if the program is running in "Network Edit" mode, the bus's property dialog box will appear as in Fig. 5-3. The circled part is added for setting the substation model related to this bus. Similar elements are also added into dialog box under "Power Flow" and "State

Estimation" mode. Enter the name of the substation model in "Substation File:" field and click the button to the right in order to open the auxiliary window. If the specified substation model file exists, it will be opened in the window; otherwise a blank substation file will be created.

2. Select the bus related to the substation model you want to edit. Then select "Analysis/Edit Substation" from the menu bar in the main window. This will activate the auxiliary window. If the specified substation model file exists, it will be opened in the window; otherwise, a blank substation file will be created. The related substation model for the selected bus in the main window will automatically appear in the auxiliary window once it is opened.

The editing procedure for the substation model, including the network elements and measurement elements, is quite the same as the editing procedure for the one line diagram in the main window. If you are not familiar with the Toolbar/Command, please refer to [1]

for instructions. After editing is over, save the substation model into a data file with an extension “.sse”.

Instead of an individual substation file, the finished model can also be saved as a common library file with an extension “.std” by choosing “File/Create LIB file” from the menu bar of the auxiliary window. Only the topological structure will be saved in this case. All the measurement information will be discarded. This library file will be used to create the data file for substations having the same or similar designs.

5.2.2 *Creation of a Substation Model Using the Library*

There are only a limited number of substation schemes in practice; hence, many substations will have the same topological structure. It is more convenient to create models for those substations via the library files. There are two ways to load the library files.

1. If you want to load only one library topology file, select “File/Load LIB File” from menu bar. Locate the file through the dialog box.
2. If you have many library files to load, put them in the same directory. Select “File/Load Batch LIB files” from menu bar. Locate the directory that stores the library files through the dialog box. The program will automatically search all the files in this directory with extension “.std” and load them.

After the library files are loaded, they will appear in the list box of the toolbox, as can be seen in Fig. 5-4. Click the type you want and the topological model of this kind of substation will appear in the window. However, the measurements need to be set manually.

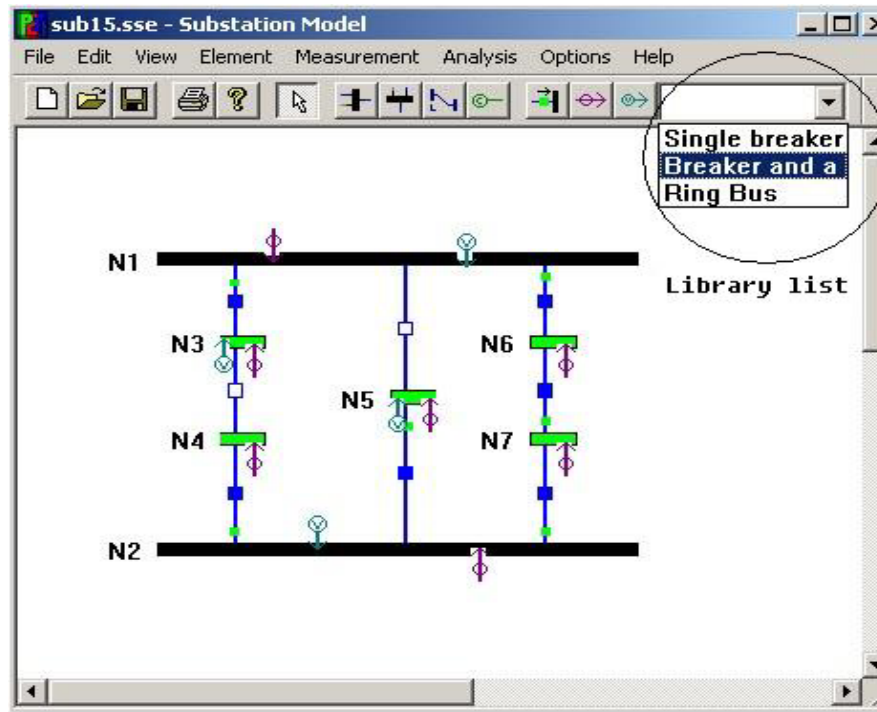


Fig. 5-4. Library file's list

5.3 Setting the Bus's Property for Related Substation Model

In order to make PET compatible with former version data file, only minimum additional data are needed. All the properties of the elements, other than the bus, remained the same. For the bus, three properties were added to link it with the substation model, as shown in Fig. 5-3.

Substation No: The unique number of the substation model related to this bus. It could be an arbitrarily value but could not be duplicated for the same system. If no substation is related to this bus, then the default value of -1 will be used.

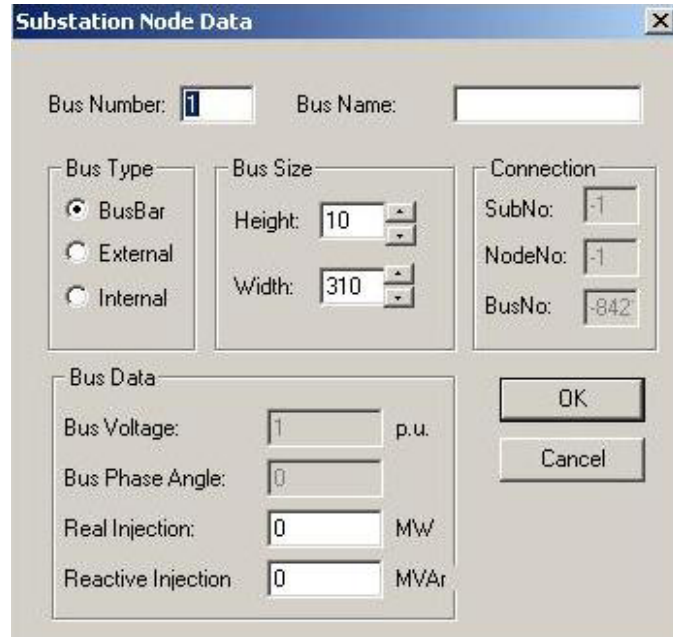
Node No: The node number in the substation model where this bus belongs. If more than one node are combined to form this bus, the node number can be chosen arbitrarily from the list of nodes. If no substation related to this bus, then the default value of -1 will be used.

Substation File: The data file name of related substation model. One bus can only have one related substation while one substation model can have more than one related bus. The program will search the same directory containing the system data file for the substation data file if no directory information is provided. If **Substation No** equals to -1 , this field cannot be edited.

5.4 Setting the Parameters of the Substation Model

5.4.1 Property of a Node

The property of a node in the substation model is quite different from the property of a bus in the main system. Fig. 5-5 shows the property dialog box for a node.



The dialog box titled "Substation Node Data" contains the following fields and controls:

- Bus Number:** A text box containing the value "1".
- Bus Name:** An empty text box.
- Bus Type:** A group box containing three radio buttons: "BusBar" (selected), "External", and "Internal".
- Bus Size:** A group box containing two spin boxes: "Height" (set to 10) and "Width" (set to 310).
- Connection:** A group box containing three text boxes: "SubNo:" (set to -1), "NodeNo:" (set to -1), and "BusNo:" (set to -842).
- Bus Data:** A group box containing four text boxes: "Bus Voltage:" (set to 1, with "p.u." to its right), "Bus Phase Angle:" (set to 0), "Real Injection:" (set to 0, with "MW" to its right), and "Reactive Injection:" (set to 0, with "MVar" to its right).
- Buttons:** "OK" and "Cancel" buttons are located at the bottom right of the dialog.

Fig. 5-5. Node's Property Dialog Box

Bus Number: The unique node number. It cannot be duplicated in the same substation model.

Bus Name: The node name. May be left blank.

Bus Type: The type of this node. There are three options. Different type of nodes will be shown in different color and have different default size.

1. *BusBar*: Represent the busbar in the substation. Will be shown in black color. Default size is: 10 for height and 200 for width.
2. *External*: Represent the nodes which have connections to other substations or have loads. Will be shown in green color. Default size is: 10 for height and 30 for width.
3. *Internal*: Represent all the internal nodes. Normally these kinds of nodes have zero power injections. Will be shown in blue color. Default size is: 10 for height and 30 for width.

Bus Size: Dimensions of the node icon.

Connection: Connection data for this node to other substations. Enabled only when the node's type is *external*.

1. *SubNo*: The substation number to which this node is connected. If there is no external connection, set it to the default value of -1 .
2. *NodeNo*: The node number which is connected to this node. If there is no external connection, set it to the default value of -1 .
3. *BusNo*: The bus number which is connected to this node in the main system. This information is very important for the substation automatic measurement generator. If there is no external connection, set it to a negative integer.

Bus Data: Relevant bus data.

1. *Bus Voltage*: The estimated voltage magnitude of this node. Only used to show the result.
2. *Bus Angle*: The estimated voltage angle of this node. Only used to show the result.
3. *Real Injection*: The active power injection for this node. Will be used in the substation automatic measurement generator.
4. *Reactive Injection*: The reactive power injection for this node. Will be used in the substation automatic measurement generator.

5.4.2 Property of a Branch

The property dialog box of a branch is shown in Fig. 5-6.

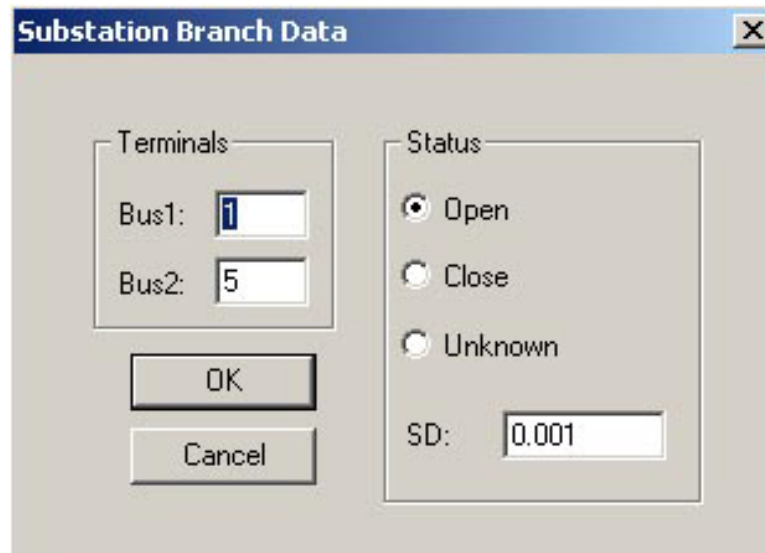


Fig. 5-6. Branch's Property Dialog Box

Terminals: Terminal node numbers.

Status: The status and its standard deviation for this branch. There are three possibilities: *open*, *closed* and *unknown*. The branch will be displayed differently for different types, as shown in Fig. 5-7. The status of a switch can also be changed by double-clicking its icon.

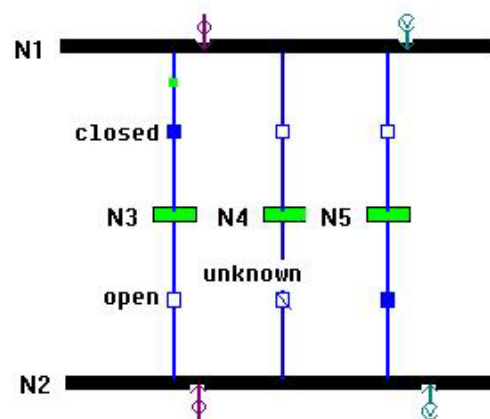


Fig. 5-7. Represents of Different Statues

5.5 The Navigator Window and Printing Function

5.5.1 Navigator Window

When the system one line diagram in the main window is too large to fit in a single screen, a navigator window (as shown in Fig. 5-2) can be used to locate an element. The thumbnail of the entire system will be shown in this window. A rectangle in red color represents the position of the current screen. Double-clicking on any part in the navigator window will bring this region into current screen. The navigator window can be shown and hidden by a shortcut key “*Ctrl+I*”. When the canvas size is large, the program will become slow with the navigator window opened. Under this situation, user can hide it to increase the performance.

5.5.2 Printing Function

Other than the generic printing function, PET provides additional settings to customize the printed page. Selecting “File/PET Printing Setup” will activate the setting dialog box. Fig. 5-8 shows the specialized setting for the printing function.

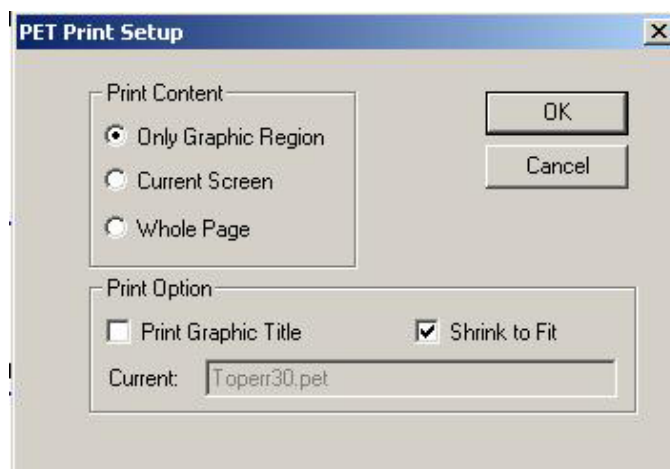


Fig. 5-8. PET Printing Setup Dialog Box

Print Content: Use this option to choose the content for printing. There are three options.

1. *Only Graphic Region:* Only print the region that contains graphical element. The blank area will be discarded.
2. *Current Screen:* Print the region that can be seen in the current screen.
3. *Whole Page:* Print the whole picture.

Print option: There are two print options.

1. *Print Graphic Title:* If this option is checked, the program will print a title for the picture. The default string for the title is the name of this file. It can be changed to any string.
2. *Shrink To Fit:* Automatically shrinks the picture to fit the whole page.

5.6 Automatic Generation of Substation Measurements

The measurement values in the substation can be automatically generated. Set the relation parameter for the main system and the substation carefully and make sure it is correct. Select “Measurement/Use the power flow result” from the menu bar. The program will run the power flow program for the main system and calculate the values for all the measurements based on the circuit breaker status and the connection information. To get the correct result, the status of all the circuit breakers should be set corresponding to the connection in the main system.

5.7 Procedure to Test the Two-Stage State Estimation

Two-stage state estimation method could be tested using upgraded PET program. The users can create their own topology error scenarios and try the two-stage state estimator to identify the topology errors by following the procedure below:

1. Create the main system

Use the main window of PET program to create the basic main system, including the network connections, generation and load setting. Place the measurements properly. After creation of the main system, validate it by the power flow function and conventional state estimation function.

2. Create the substation model

Use substation model introduced in section 3.4 to represent the actual substation. Create the substation data file for specified buses using the auxiliary window. Do not need to create substation data files for all the buses. Only the substation models in the interested area are needed.

3. Set the correct measurement values based on the correct network

Make sure the connection information for those buses in the main system and nodes in the substation models are correctly set. The statuses of the circuit breakers must be corresponding to the current connection of the main system. In the main window, select “Measurement/Use power flow analysis results” to set the correct values for the measurements in the bus/branch model. For each concerned substation, in the auxiliary window, select “Measurement/Use power flow analysis results” to set the correct values for the measurements in the current substation model.

After this step, select “Analysis/Run SSE Estimation” to check whether the measurements and connections are correctly set. If so, the program should not generate a topology error alarm.

4. Set up the topology error

There are four possible types of topology errors (section 3.7). For *type3* and *type4*, simply add a circuit breaker to the corresponding transmission line and change the status of this circuit breaker to be wrong. For *type1* and *type2*, a pseudo-branch (impedance will be set to $j10^{-5}$) is added between those buses and a circuit breaker is added in this branch. Change the status of this circuit breaker to represent a merged case and a split case.

For all those cases, just change the status of corresponding circuit breaker to introduce a topology error.

5. Run the two-stage program

After introducing the topology error, select “Analysis/Run SEE Estimation” to load the two-stage state estimator. After the first stage, if there is topology error detected, system will show the index of suspect buses and ask whether to run second-stage state estimation. Click “yes” to continue. After the estimation is over, if there are mismatches between the assumed status and estimated status of circuit breaker included in those suspect substation, system will use message boxes to show the location of the topology error.

An example will be shown in the following section.

5.8 Examples

The numerical example shown in section 3.8.1 is used to illustrate the user interface. Here is the summary of this scenario. This case is based on the IEEE 30 bus system (Fig. 5-9). We suppose bus 16 and bus 19, which are circled in Fig. 5-9, belong to the same substation.

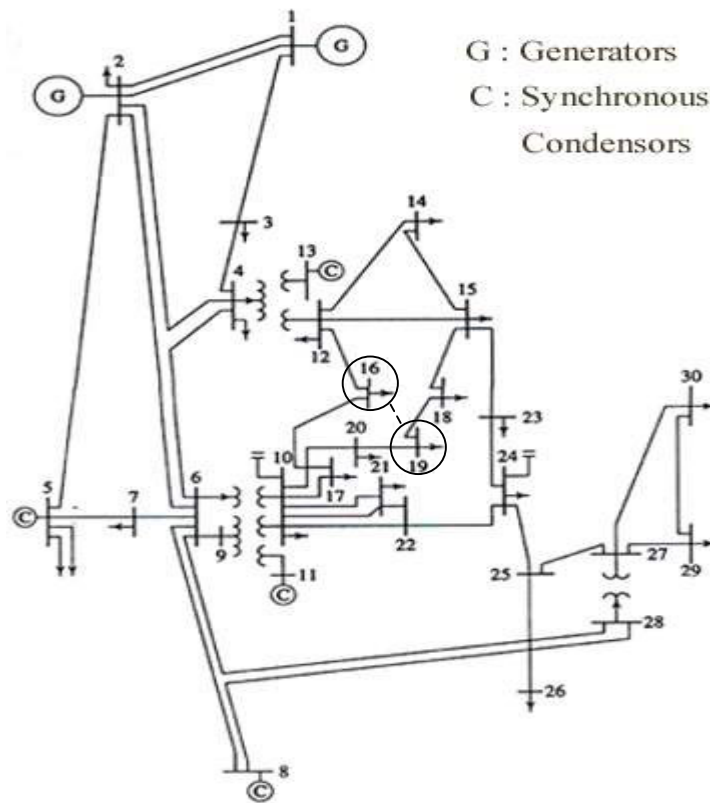


Fig. 5-9. IEEE 30 bus system

Fig. 5-10 shows the detailed topology of this substation with the status of the circuit breakers. The scenario of this demo is described as the following:

- The true status of the *CB N5-N2* is open, which makes this substation appears as two split buses in the bus/branch model; and
- Assume that the operation of the *CB N5-N2* is not reported to the control center. The control center still assumes the status of *N5-N2* as closed. Then this substation will appear only as one bus in the bus/branch model.

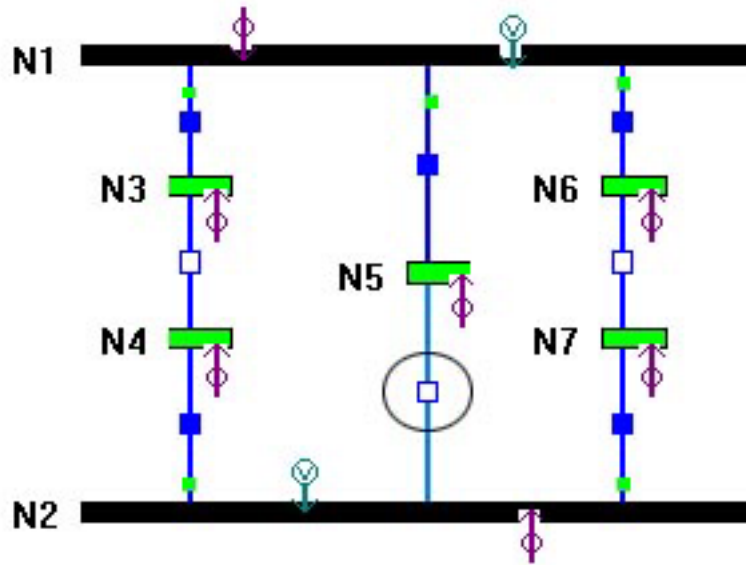


Fig. 5-10. Detail model of Substation 16

A system model corresponding to Fig. 5-9 was created in the main window of PET and the detailed substation model for Fig. 5-10 was created in the auxiliary window (*step1 and step2 in section 5.7*). After this, we get the correct connection and correct topology for the concerning substation, which can be seen in Fig. 5-11.

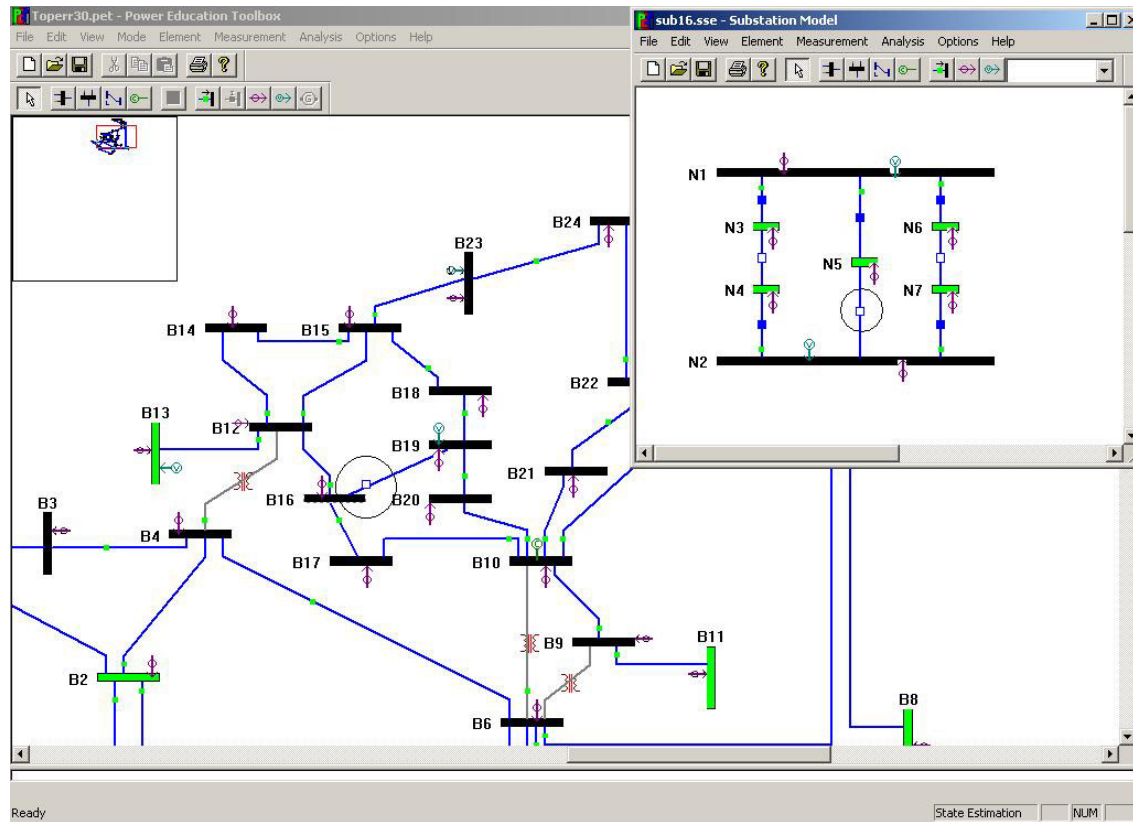


Fig. 5-11. Correct topology for the test system

We can see from Fig. 5-11 that the circled switch between bus 16-19 in the main window and the circuit breaker between N5 and N2 in the auxiliary window are open, which is the correct status of this case.

The measurements also were placed as shown in Fig. 5-11. The values of all the measurements can be automatically calculated by the power flow function and the substation measurement automatic generator. For further details, please refer to [1] and section 5.6 for instructions.

After validating the system, the substation model and the data by the conventional state estimation, we need to introduce the topology error manually. This could be done by closing the circled pseudo switch and circuit breaker. Fig. 5-12 shows the diagram after introducing of the topology error.

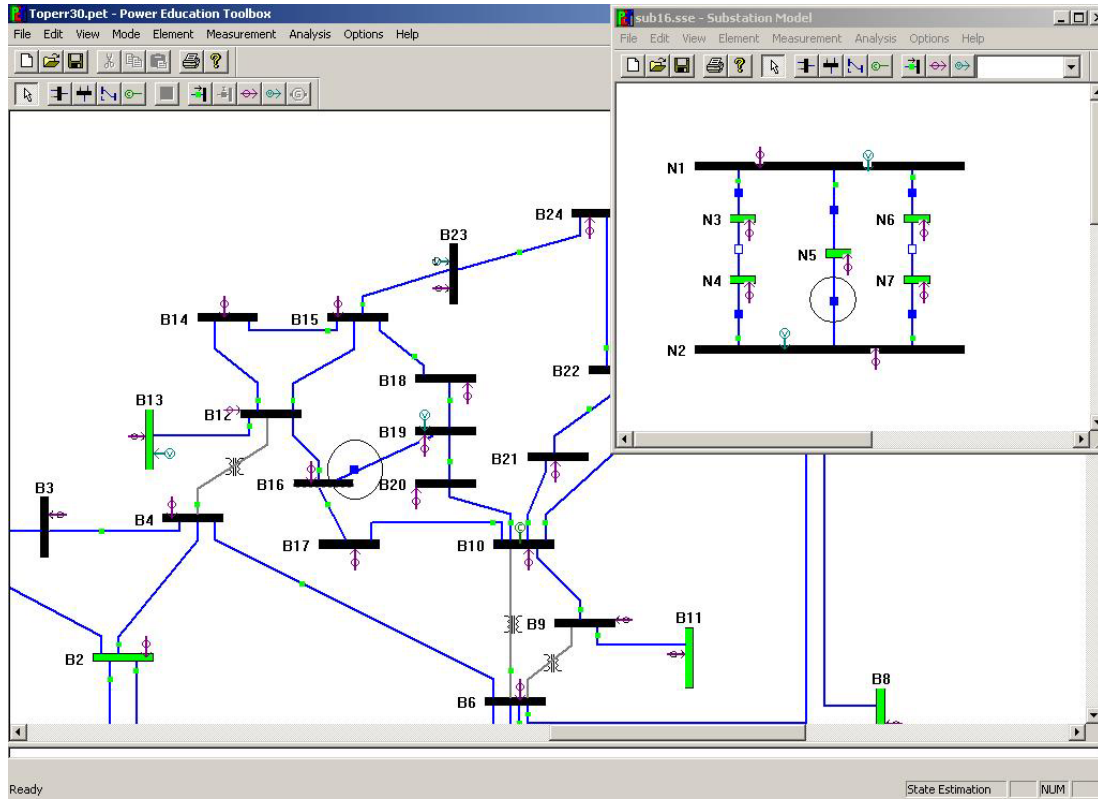


Fig. 5-12. Wrong topology for the test system

Select “Analysis/Run SSE estimation” to launch the two-stage state estimator program. After the first stage, there will be a warning message indicating that there is a suspicious bus (16) as show in Fig. 5-13.

Clicking “OK” will initiate the second-stage state estimation. After this, the normalized power flows through all the circuit breakers will be checked and the statuses of them will be estimated. PET then compares the estimated statuses and the pre-set statuses. A mismatch will be found in the circuit breaker from N5 to N2. A warning message, as in Fig. 5-14, will warn the user about the identified topology error.

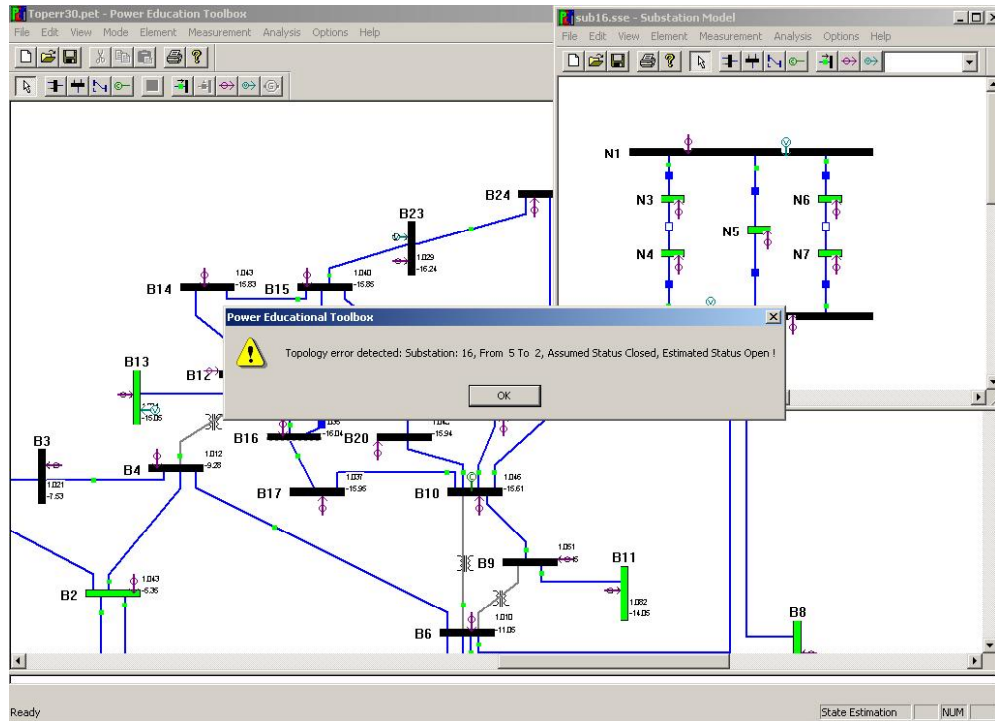


Fig. 5-13. Diagram after the first stage

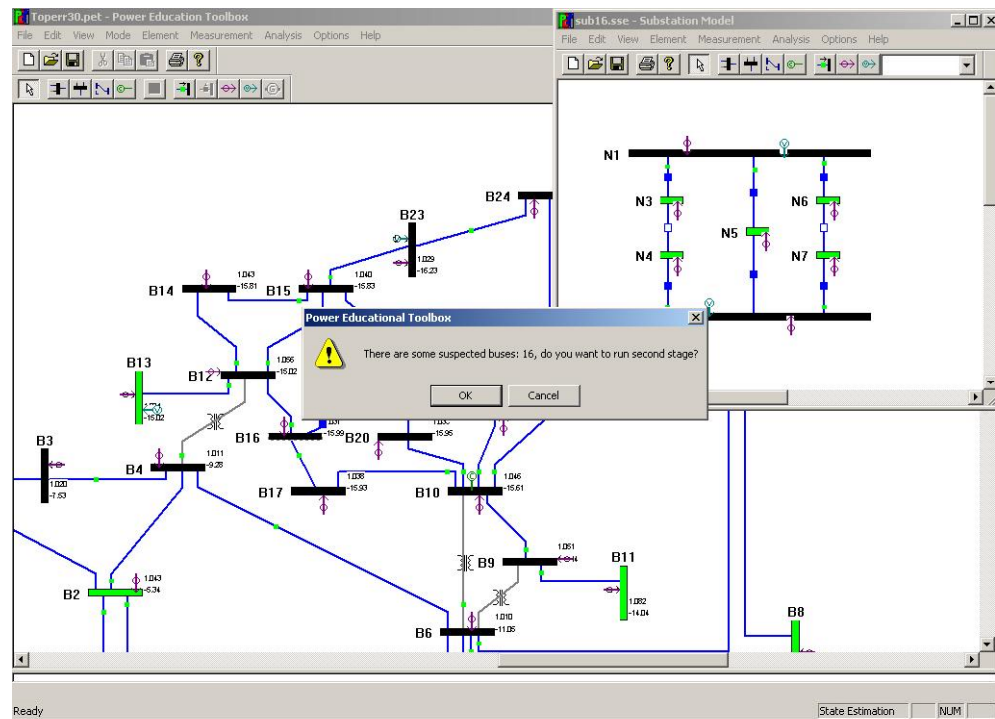


Fig. 5-14. Diagram after the second stage

5.9 *References*

- [1] A. Abur, "Power Education Toolbox (P.E.T) User Manual", Texas A&M University.

6. Graphical User Interface for the Substation Processor

This section instructs user how to operate the Substation State Estimation Software and explains its features.

6.1 Graphical User Interface (GUI)

As the user starts software by typing **sses** at the prompt in the Matlab command window, Graphical User Interface (GUI) opens. The screen that is presented to the user is shown in Fig. 6-1.

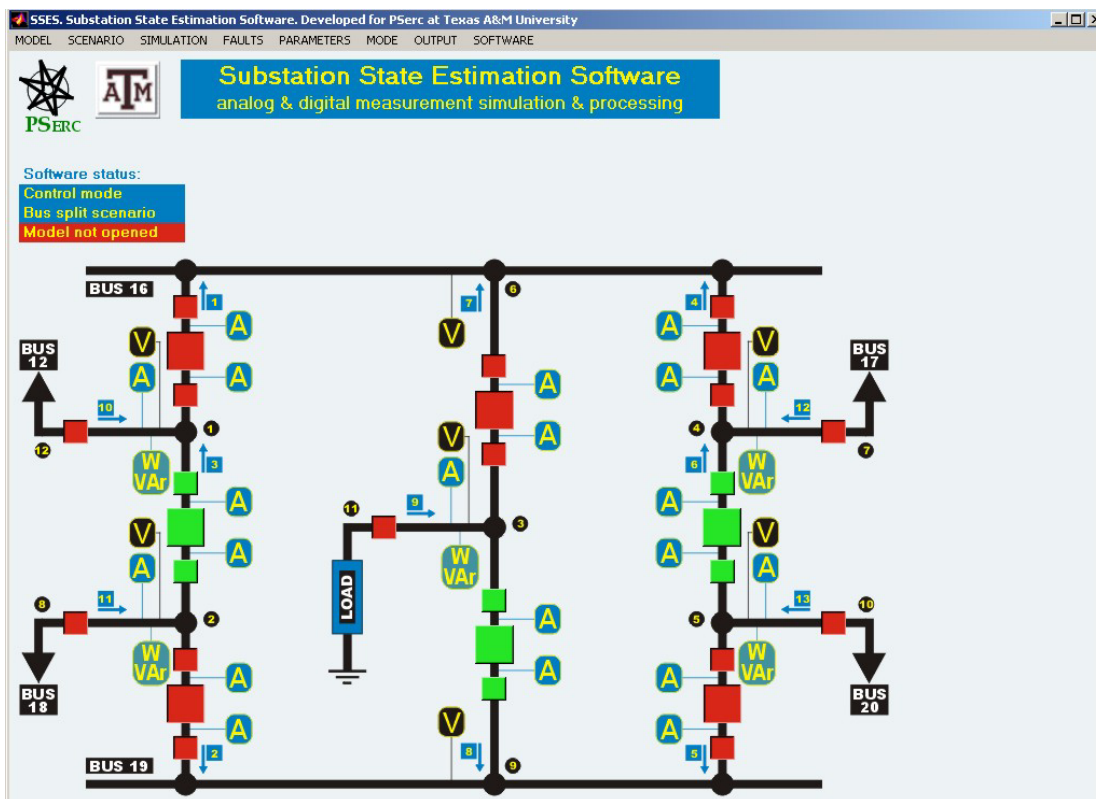


Fig. 6-1. Graphical User Interface (GUI) after the software is started

The interface presents a single-line diagram of the substation that is modeled and utilized throughout the software operation. As the software simulates the computer in the substation control room, the diagram can be viewed as a visualization of the substation switchyard.

The substation consists of two busbars (designated as **BUS 16** and **BUS 19**) where four transmission lines and a load are connected. Transmission lines lead to other buses in the IEEE 30 bus power network (**BUS 12**, **BUS 17**, **BUS 18** and **BUS 20**). The load is locally grounded. The user can see the type of connections. Switch devices are presented as red and green squares. Larger ones are circuit breakers while smaller ones are disconnect switches. Their current state is reflected through their color:



OPEN SWITCH



CLOSED SWITCH

Distribution of metering equipment around the substation is shown as well. Instruments are:



A-meters



V-meters



combined W- and VAr-meters

Following color convention for measurements is used in the interface:

- Blue color A-meters and associated measurements (also branches)
- Black color V-meters and associated measurements (also nodes)
- Green color W- and VAr-meters and associated measurements.

Since nodes and branches are referred to in most processing reports, the user needs to recognize their appropriate labels:



Branch number (within blue square) and orientation (blue arrow)



Node number (within black circle)



On the left hand side above the substation diagram, the software status box displays the status of three software features: current mode of operation, initial topology scenario, and status of substation model. Blue color is used to describe normal operation, while red color messages convey alert information. Mentioned features will be explained in detail later.

Other parts of the user interface screen:



SSES window title mentions PSERC (Power System Engineering Research Center), which is an industry and university consortium that funded the project.

In addition, Texas A&M University is mentioned and both TAMU and PSERC logos are displayed in the upper left corner of the user interface.



SSES screen title is positioned above the substation diagram.

Very important part of the GUI is the menu. This drop down menu is used for most software control activities. By clicking anywhere on the menu, depending on the position of the mouse cursor, an appropriate menu will drop down and offer a list of associated activities. Top level menu options are shown in Fig. 6-2. All of them will be described next.

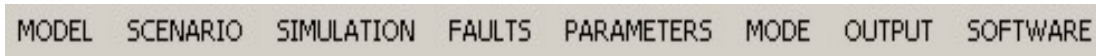


Fig. 6-2. SSES GUI menu

6.2 User's Guide on Running the Simulation

When the software is started and the initial user interface screen is displayed, user needs to perform several steps to make the simulation run. All the functions on the initial user interface screen are disabled at first. There are only a limited number of operations that can be done through the menu. This functionality was chosen because after software initialization, the substation model is not automatically opened and there is not much to do without the model or associated measurements. Software status box displays following warning: **Model not opened**.

Normally, the first step after software initialization is opening the substation model. This is done through the user interface menu. Menu options are ordered from left to right in the sequence that they are most commonly accessed. From that prospective, **MODEL** is the first menu option on the left hand side. User needs to click on **MODEL** → **Open** in order to open the substation model. A new window with SIMULINK substation model will be opened, but the screen will automatically switch back to the GUI. The model will exist in the background and the user is not supposed to perform any actions on the model directly in

the model window. The model needs to be controlled strictly through the GUI. Even manually closing the model window can permanently change the model settings, which would later cause certain software malfunctions.

When the substation model is opened, the software status box displays the following message: **Model opened**. The model can be closed through the user interface menu by clicking **MODEL** → **Close**. At this point, such an operation would revert the software to the previous state, which is not something that we want to do. Actually, once opened, the substation model does not need to be closed at all throughout the software operation. It is even simpler, at the end of the work, to close both the model and whole software with **SOFTWARE** → **Exit** menu command.

There are two initial scenarios implemented in SSES: Line outage and Bus split scenarios. One of them needs to be employed before the simulation is run. Scenarios actually determine the initial substation switch devices state and associated network equivalent that is used in the model. Difference between scenarios regarding the switch devices state is shown in Fig. 6-3.

Bus split scenario is the default scenario upon software initialization. It can be kept or changed. It assumes independent operation of two substation buses. Line outage scenario additionally assumes transmission line 12-16 disconnected.

Initial scenario can be changed either when the substation model is opened or closed, but certainly before the simulation is run. Scenario change can be done through user interface menu: **SCENARIO** → **Line outage** and appropriate substation switch devices will change their status. In addition, software status box will read accordingly

Line outage scenario.

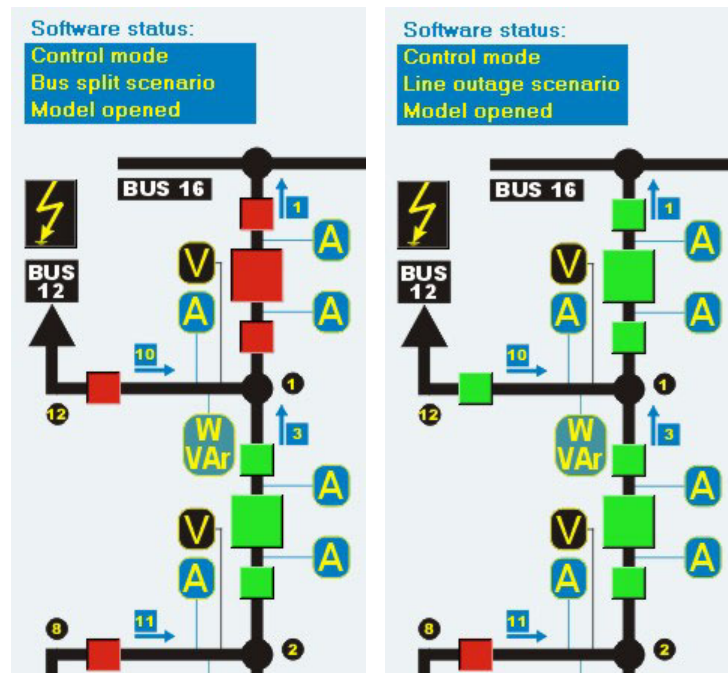


Fig. 6-3. Two initial scenarios

Similarly, scenario can be switched back to the default one by clicking on the menu **SCENARIO** → **Bus split**. It would also be accompanied by change of appropriate switch element statuses and the software status box will display **Bus split scenario**.

When the substation model is opened, substation switch device control is enabled through the GUI. In other words, the user can change the status of any switch element in the substation model by simple clicking on an appropriate switch element square on the user interface screen. The switch elements are presented as green or red squares depending whether they are opened or closed respectively. Clicking the switch square will change its status and color accordingly. The corresponding switch element in the substation model (that is opened in the background) will receive the new status information from the GUI. All this holds for the Control mode while other modes can have slightly different logic. Software modes of operation will be discussed later in detail.

Changing the status of switch elements is associated with switching sequences. the basic switching sequence rule is that a disconnect switch should never be opened while the circuit breaker is closed. This is a consequence of the disconnect switch's inability to break current (i.e., to extinguish electric arc). Operating the disconnect switch while the associated circuit breaker is closed can be dangerous both for personnel and equipment. SSES has this switching sequence rule implemented. An attempt to open the disconnect switch before the circuit breaker will result in a warning and the action will be cancelled. The warning that is generated (for branch 1) is shown in Fig. 6-4..

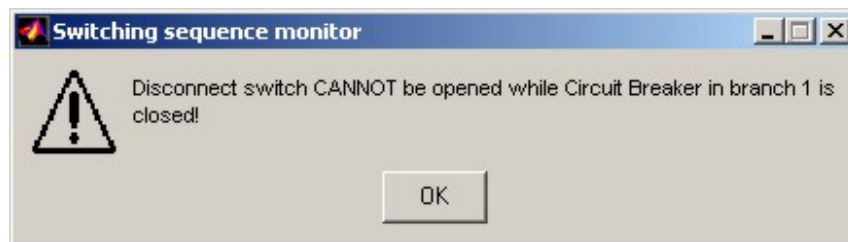


Fig 6-4. Switching sequence monitor warning

One more thing that appears when the substation model is opened is fault pushbutton. As it can be seen on the GUI screen, seven fault buttons are located next to each transmission line, load and busbar. Fault pushbutton is shown in Fig 6-5.

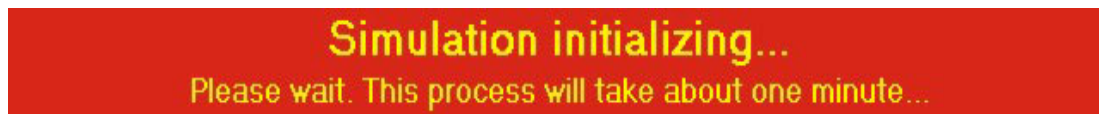


Fig. 6-5. Fault pushbutton

The purpose of those buttons is to control opening two or more circuit breakers at a time. As it happens when the fault is detected anywhere in the system, all circuit breakers closest to the fault need to be opened in order to clear the fault. Since the transmission lines in this substation layout are associated with two circuit breakers, they both need to be opened upon the fault. (The opposite terminal is not considered here). There is a similar situation with load. Busbar faults are cleared in the same manner, except here we have three circuit breakers associated with each busbar. They all need to be opened upon the bus fault. Fault pushbuttons are meant to be used in simulating the fault clearing.

Clicking on the fault pushbutton will open all associated circuit breakers. If some of the circuit breakers are already opened, their status will not be changed. Instead of clicking on the fault pushbuttons, the same effect can be obtained through the user interface menu: choose the **FAULTS** option and then click on the corresponding element (**line 12-16**, **line 18-19**, **line 17-16**, **line 20-19**, **Load**, **Busbar 16** or **Busbar 19**).

Finally, we come to the running of simulation. The simulation is started from the user interface menu: **SIMULATION** → **Start**. The first thing before the simulation starts runs is a process of substation model initialization. This must be performed by SIMULINK whenever the model is employed. It takes a certain amount of time to check the connections in the model and accomplish other important tasks before the model is run. In the case of the substation model, the initialization process lasts up to one minute (depending upon the computer processing power). During that time, this appropriate message is displayed on the screen:







When the initialization process is done, the simulation starts running. New boxes appear on the user interface screen. They are used as soon as the first snapshot of measurements is being processed. Further on they are constantly used to display snapshot processing results.

In the lower right corner, the snapshot counter will appear. It starts from zero when the simulation is started and increases by one whenever a new snapshot is being processed. It also alternates the color of the snapshot number (black and red) in order to emphasize when new data has been displayed. An example of snapshot counter is shown in Fig. 6-6.



Fig. 6-6. Snapshot counter box

Right above the snapshot counter is a pushbutton that can be used to control the simulation (i.e., to alternatively pause and continue simulation when clicked). Accordingly,







the button will display the following:  or . The simulation status is reflected through the software status box that displays  or  as an appropriate simulation status message.

Full simulation control is achieved through **SIMULATION** user interface menu:

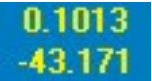
- While the simulation is running, it can be either paused (**SIMULATION** → **Pause**) or stopped (**SIMULATION** → **Stop**).
- While the simulation is paused, it can be either continued (**SIMULATION** → **Continue**) or stopped (**SIMULATION** → **Stop**).
- While the simulation is stopped, it can be run (**SIMULATION** → **Start**).


Appropriate simulation status is always reflected through the software status box. After the simulation is stopped, it needs to go through initialization process whenever it is run again. Continuing the simulation after being paused does not require initialization.

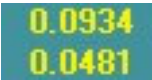
Snapshot processing results are displayed on the user interface screen. There are three types of messages being displayed: certain algorithm outcomes, measurements and processing reports.

Algorithm outcomes are described earlier when the algorithms were analyzed. Those are represented as icons next to the appropriate measurements, switch elements or nodes. Icons are as follows: , , , ,  and . They have all been already explained in section 3.3.

Values of analog measurements are displayed in corresponding measurement boxes after each snapshot is being processed. Instruments are associated with colored boxes:

 Blue boxes are reserved for current measurements. Upper value is phasor magnitude and the lower one represents its phase angle.

 Black boxes are reserved for voltage measurements. Upper value is phasor magnitude and the lower one represents its phase angle.

 Green boxes are reserved for power flow and power injection measurements. Upper value is active power and the lower one represents reactive power.

All analog measurements are normalized and are in relative units. Status measurements are displayed next to the branch switch devices (or groups of switch devices). Switch elements can be operated (opened or closed) through the user interface during the simulation run. This will cause change in status and analog measurements as soon as the next snapshot is being processed.

After each snapshot processing is finished, processing reports are displayed along the right edge of the user interface screen. Most recent reports are displayed on the top after all previous ones are being shifted downwards. The number that the report begins with represents corresponding snapshot number. All possible reports are already discussed when the processing algorithms are being analyzed in section 3.3.

Simulation, model and processing parameter can be accessed through the user interface menu **PARAMETERS**. Actually, changing the simulation parameters (mode solver or simulation time) is planned for future software versions and it is disabled in version 1.0.

Substation model parameters can be accessed through the user interface menu **PARAMETERS** → **Model** → **Switch**... The dialog box that is opened is shown in Fig. 6-7.

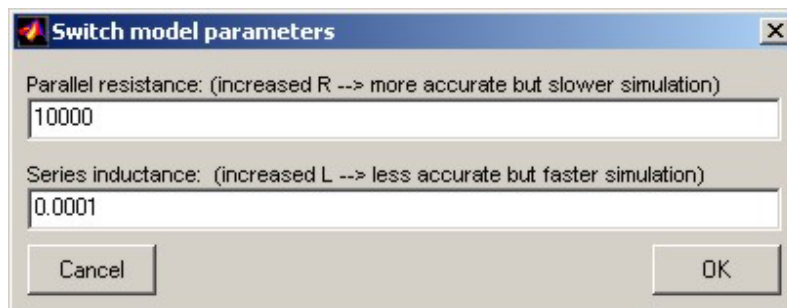


Fig. 6-7. Dialog box for adjusting the switch model parameters

As it was mentioned in section 2.2, the switch model needs parallel resistance and series inductance to operate correctly. In an ideal case, those values should be infinite resistance and zero inductance. Unfortunately, the model does not accept those ideal values. Therefore, resistance needs to be some finite number and inductance needs to be larger than zero.

Non-ideal values introduce errors in simulation. Actually, there is a trade-off between simulation speed and accuracy. The closer the parameters are to the ideal values, the slower the simulation runs. Default values are empirically chosen to make the best trade-off. The user can increase the value for parallel resistance and/or decrease the value for series inductance and that will make the simulation be more accurate but slower. On the contrary, it is possible to sacrifice accuracy in order to gain on simulation speed.

The main processing routine parameters can be accessed through the user interface menu by selecting **PARAMETERS** → **Processing**.... The dialog box that opens is shown in Fig. 6-8.

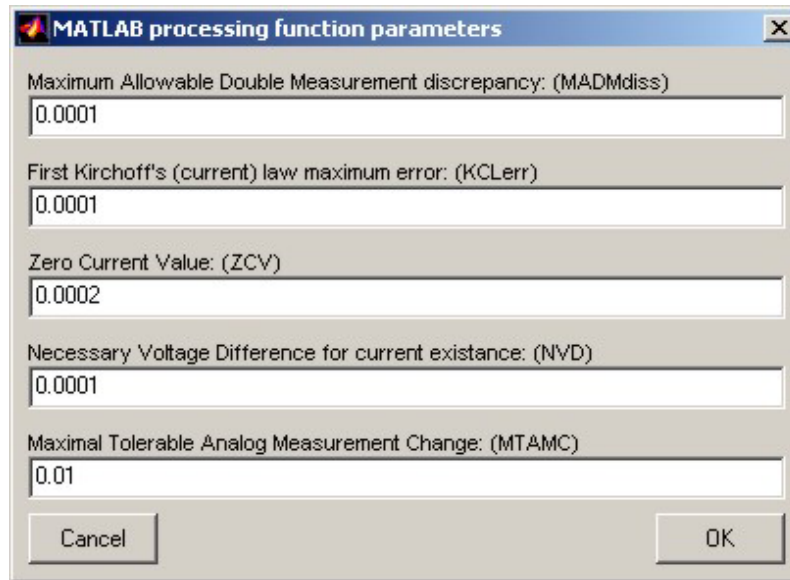


Fig. 6-8. Dialog box for adjusting the processing parameters

Dialog box contains all the processing parameters in a single box. One or more of them can be changed before the OK button is pressed. The meaning of each of those parameters has been explained in section 3.3 where the processing algorithms were discussed. If the dialog box is opened during the simulation, the execution of simulation is paused as long as the OK or CANCEL button is pressed.

The last thing regarding the parameters is restoring the default values. It can be done through the user interface menu by selecting **PARAMETERS** → **Default**. It will reset all the parameter values back to their default values written in software.

An example of Graphical User Interface screen during the simulation execution is shown in Fig. 6-9. All the previously discussed details can be seen.

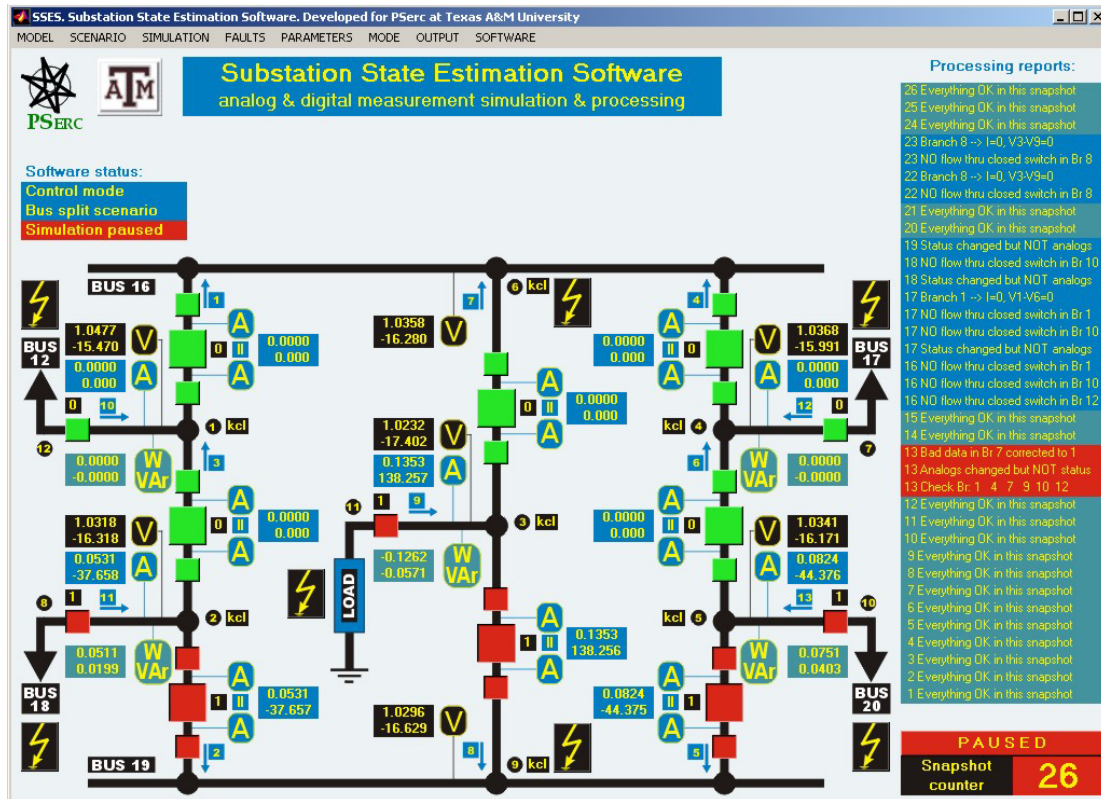


Fig. 6-9. An example of Graphical User Interface screen during the simulation execution

6.3 Modes of Operation

Software is designed to operate in several modes. The main mode that software is operating in by default is Control mode. As soon as the software is started, software status box displays following software mode message: **Control mode**.

The control mode relates to the substation operation when the user can change the status of switching devices. The user “controls” the operation of the substation by switching in and out certain transmission lines or loads, or performs flow transfers from one busbar to the other. Transmission line and busbar fault clearing can be simulated as well by clicking the fault pushbuttons. During all these control operations, the simulation is running and the substation state estimator acquires measurements and performs their processing. The results of processing are readily displayed on the user interface screen. The user can promptly see how certain control actions influence the power flows and other quantities in the substation.

In order to present the behavior of the processing algorithms in the presence of erroneous status measurements, the Bad data mode was devised. The user can switch to the Bad data

mode through the user interface menu by selecting **MODE** → **Bad data**. The software status box will reflect the change through the message: **Bad data mode**.

In the Bad data mode, the user can click on the switch element squares to change their status. The big difference between the Bad data mode and the Control mode is that the change of status will not influence the substation model, but only the processing. In other words, when the user changes the status of certain switch elements, the only thing that happens is that erroneous data is introduced. No status change occurs within the substation model that is simulating. Therefore, analog measurements that are coming from the model remain the same (and correct) whereas affected status measurements become erroneous. This way user creates bad data.

The processing algorithms are supposed to detect and eliminate bad data measurements. The processing algorithms are the same as ones applied during the control mode (i.e., the algorithms “do not know” whether the software is in the Control or the Bad data mode).

An erroneous status of a switch element that is reported to the processing routine is designated by the bad data symbol “BD” on the switch element button. An example of a switch element that is closed, but, due to bad data, appears as if were open is represented as



. In the opposite case, the switch element that is opened, but due to bad data it appears as if it were closed is represented as



When the processing algorithms detect bad data, corrected status is displayed next to the appropriate switch element. The background of corrected status is red. In addition, processing report is generated for each eliminated bad data. Therefore:

- For status **1** an example of generated report is **3 Bad data in Br 7 corrected to 1**,
- For status **0** an example of generated report is **11 Bad data in Br 8 corrected to 0**.

The user can always return to the Control mode by selecting **MODE** → **Control** from the user interface menu. The software will automatically eliminate all bad data that were introduced during the Bad data mode. In other words, statuses of switching elements will be reverted to their true values.

Another software mode that can be selected is Demo mode. In this mode, the software demonstrates certain predetermined switching actions and explains their outcome. The mode is started from the user interface menu by selecting: **MODE** → **Demo**. Software status box displays mode status as **Demo mode**. Since the software confers certain information to the user during the Demo mode, a special demo banner is

opened above the substation layout on the user interface screen. Introductory demo banner is shown in Fig. 6-10:

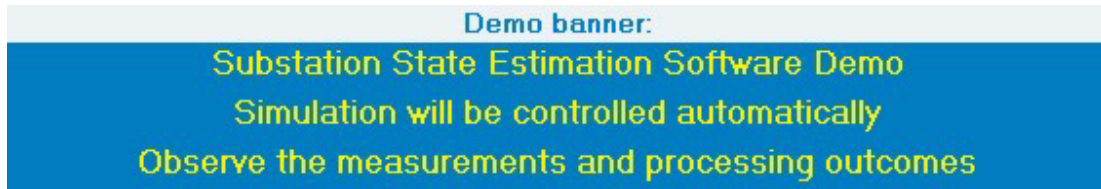



Fig. 6-10. An example of demo banner

In the Demo mode software takes control over the simulation. Most of the options within user interface menu are disabled. The user is to observe the changes upon switching actions done by the software. The user can pause and continue the simulation in any moment if additional time is required to analyze the outcomes.



The Demo mode can be stopped only by clicking on  button located between software status box and demo banner. It can be done at any time. The software will stop the simulation and revert to the Control mode. The substation model needs to be closed before the Demo mode can be started again.

The last mode that is not implemented in this version of the software is Transitions mode. Upon selecting **MODE** → **Transitions** from the user interface menu, the message that is displayed is shown in Fig. 6-11. The Transitions mode is in process of development and will be included in future software versions. It will check switching sequences and the path of substation transitions.

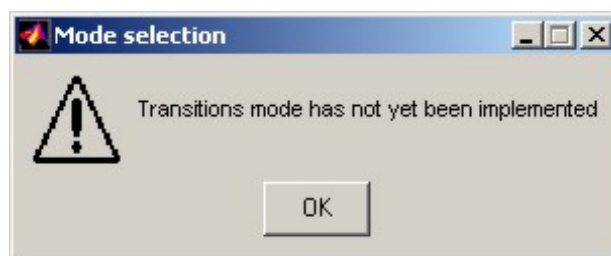


Fig. 6-11. The warning when attempted to change the mode to the Transitions mode

6.4 *Software Components and Installation*

Substation State Estimation Software is delivered as a single ZIP file (**SSES version 1.0.zip**). This is done since the software consists of 122 files distributed in 5 folders. Zipping those files facilitates convenient storing and handling all the files without possibility of losing any single file.

Part of the software installation requires extracting files from the software ZIP file. Extraction can be done by double-clicking the ZIP file (with assumption that WinZip software is previously installed on the computer). Extraction will be completed with original folder structure preserved. In other words, after the software files have been extracted, new folders will be created and each would contain appropriate software files.

The top folder is named **SSES** and it contains four subfolders and five files.

Files are:

- **DemoStop.m** Routine that stops execution of software demo
- **SSES.m** Main software routine
- **SSESdemo.m** Software demo routine
- **SSESprocessing.m** Main processing routine
- **SSEM30.mdl** Substation model file

Subfolders are:

- **data** Contains 5 mat files with GUI images and network equivalents
- **menu** Contains 26 m files that are user interface menu routines
- **shot** Contains 57 m files that capture measurement snapshots
- **toggle** Contains 29 m files that control switch element toggle behavior

Since the SSES is Matlab based, it is assumed that the computer has Matlab software already installed. At least Matlab version 5.3 (release 11) and associated Simulink tools with Power blockset is required. After software extraction, one more thing needs to be done as a part of software installation. All five newly created folders need to be added to the Matlab path browser list.

Software is started by typing **sses** at the prompt in the Matlab command window. This is the only file that user needs to run by typing at the command line. Complete software control is further achieved through the user interface.

6.5 Example of Software Outputs

The results of processing are displayed on the user interface screen for each snapshot of data. The user can see the values of processed measurements and a recent history of processing reports. Displaying results on the screen is just an auxiliary way to output the results.

Substation State Estimation Software is designed to operate on computers in the substation control room. One of the main software functions is to communicate processing results to the power system state estimation center. This task is accomplished through the exchange of data files that contain list of analog and digital measurements. Data files can be communicated on a regular basis after each snapshot but such a rate may be too frequent. The other option is to leave the power system control center to demand data files when necessary.

There are two types of data files: constant topology and measurements. Constant topology data file contains list of nodes with their classification and list of branches with their “from” and “to” nodes. This file describes constant topology data; i.e., data that does not change in time and does not depend on simulation execution.

The software creates a constant topology data file at the user’s request at any time. Since this file does not depend on the simulation, it can be created even before the simulation is run. To request this output file, the user needs to select from the user interface menu: **OUTPUT** → **Topology**. The message that is displayed is shown in Fig. 6-12.

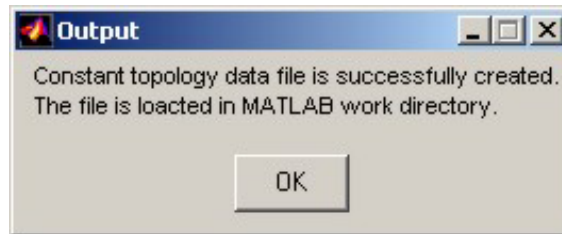


Fig 6-12. Outputting the topology data file

Constant topology data file is named **Sub1619.top**. It is created and saved in the MATLAB work directory. It can be opened by text editor (e.g. notepad). An example of constant topology file is shown in Fig. 6-13.

```

12
1 3
2 3
3 3
4 3
5 3
6 1
7 2
8 2
9 1
10 2
11 2
12 2
13
1 6
2 9
2 1
4 6
5 9
5 4

```

Fig. 6-13. .Top file

The other data file contains processed analog and status measurements required by the overall state estimation computer. The data in this file is usually different from snapshot to snapshot. The file can be created by the user through the menu: **OUTPUT** → **Measurements**. The message that is displayed is shown in Fig. 6-14. Since the measurements depend on the substation model simulation, they are available only when the simulation is running. Therefore, the user can request that the measurement data file be created only when at least one snapshot is being processed. Premature request would result in a warning shown in Fig. 6-15.

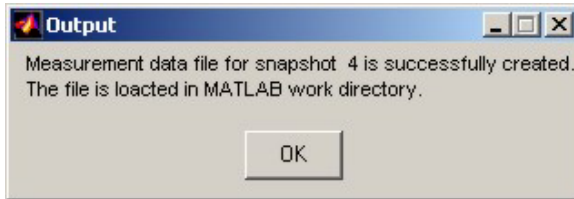


Fig. 6-14. Outputting the measurements data file

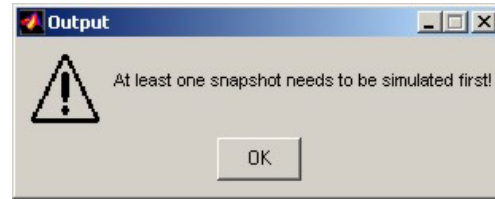


Fig. 6-15. Warning due to premature request

The name of measurements data file consists of a part same as constant topology data file, word “shot” and the snapshot number. The extension is “.sub”. An example of measurements data file name is **Sub1619shot4.sub**. The file is also created and saved in the MATLAB work directory and it can be opened by text editor (e.g. Notepad). An example of measurements data file is shown in Fig. 6-16.

The first line in the measurement data file is actually the name of the corresponding constant topology data file. Then there is a list of neighboring substation numbers.

Next is a list of status measurements (i.e. variable topology data – list of branch statuses). Analog measurements are listed in the follow-up as voltages, power flows and power injections. Each section has the number of measurements first, then values and corresponding standard deviations.

Format of the output files is understandable to the overall state estimation computer. Software has the data files communicated on the user’s request, whereas in the real application it would be sent on the upper level request.

Substation State Estimation Software operates with more data then it is communicated to the power system control center. All the redundant data and data that is used temporarily may be of interest for other substation functions. Substation computer could serve several different tasks within the substation with minor increase of the processing power.

```

Sub1619.top
17 99
18 99
20 99
0 99
12 99
1 0.001
1 0.001
0 0.001
1 0.001
1 0.001
0 0.001
1 0.001
0 0.001
1 0.001
0 0.001
1 0.001
1 0.001
1 0.001
1 0.001
1 0.001
7
1 1.03759 -15.9248 0.0010
2 1.03947 -15.9767 0.0010
3 1.02591 -17.2664 0.0010
4 1.03374 -16.2863 0.0010
5 1.04035 -15.9180 0.0010

```

Fig. 6-16. .Sub file

The last thing to be mentioned is the software about box. It opens up when the user selects **SOFTWARE** → **About** from the menu. The box is shown in Fig. 6-17.

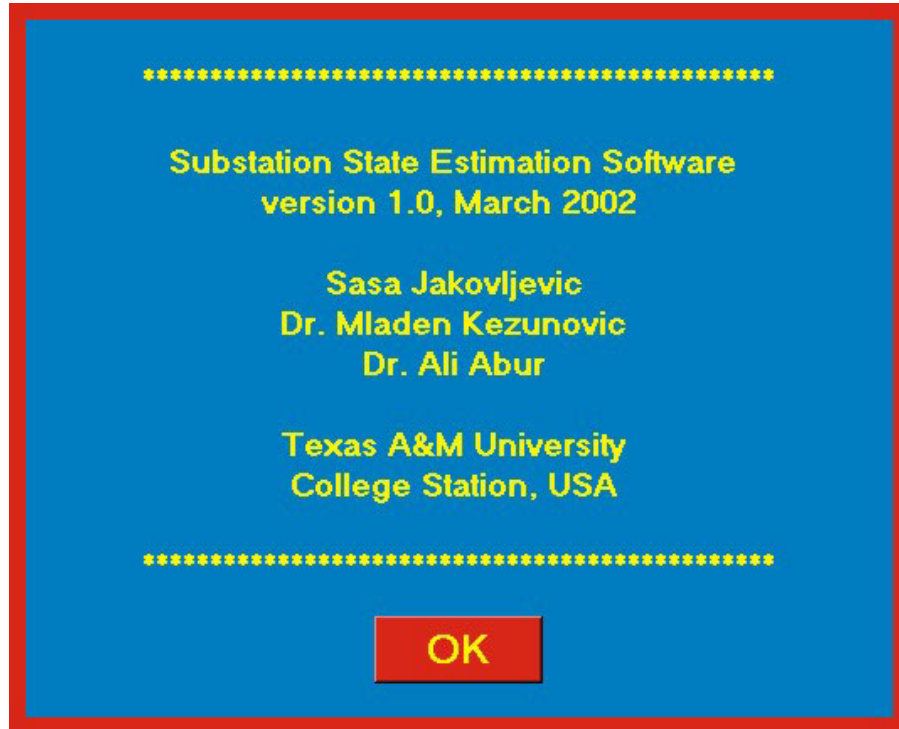


Fig. 6-17. The software about box

7. Conclusions and Future Work

7.1 Project Accomplishments

This project produced several results, computer software and associated user interfaces, as well as their associated documentations. The following is a summary of the analysis results and developed prototype software in this project.

- Substation measurement simulator and processor: A Matlab-based software tool which allows multi-phase simulation of detailed substation operation, including breaker and tie switch operations, faults and incorrect breaker status scenarios. The program also provides a substation level measurement pre-processor which receives raw digital samples from various transducers in the substation, executes several consistency checks, and generates the substation measurement data base to be transmitted to the control center.
- Two stage state estimator: A LAV state estimator with two stages. The first stage results are processed in order to determine suspect substations if any exist. The second stage identifies the actual topology error at the substation level for the suspected substations. The detailed substation measurements are supplied by the substation processor for the suspect stations.
- A method for identifying suspect substations: An improved method for identifying the suspect substation after the first-stage state estimation is developed. A library of cases for testing alternative identification methods' performance is also developed.
- User interface development: All software developed in this project are equipped with associated user interfaces, facilitating the testing of these prototype programs is easy.

7.2 Conclusions and Future Work

The results of this project strongly support the viability of utilizing substation level measurement processing for improving topology error identification function. The developed algorithms and associated software demonstrate that a two level formulation of the topology error processing is not only possible within the existing capabilities of the substation intelligent electronic devices (IED) but also quite desirable. The details of pre-processing of the substation level measurements are presented. An improved procedure for identifying the suspect substation after the first-stage state estimation is proposed and its performance is verified using a library of test cases. This library is custom built and contains fifty cases of different topology error types. The project also investigated

other sources of error such as those due to three phase unbalances and presented a sensitivity study for the estimated states with respect to the system unbalances.

The project has so far concentrated on the positive sequence state estimation and the associated topology error problem. However, the unbalance sensitivity study reveals that potential bias and shift exist in the state estimates when the system operation is not balanced. Hence, the project can be extended to consider the three-phase state estimation problem and related issues. Another issue that can be addressed is the choice of measurement weights, which are currently assigned in an ad hoc manner. This extension, however, requires actual historical data that can then be processed to extract proper weights for individual measurements.