Sensor Placement for Enhanced Distribution Protection

Prepared by[*]

K. Tomsovic R. Blakely F. Soudi
School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA 99164-2752

Project Managers
K. Lau and J. Yee
Research and Development Department
Pacific Gas and Electric Company
3400 Crow Canyon Rd.
San Ramon, CA 94583

July 2, 2000

---

[*] Project support under PG&E contract no. Z-19-2-410-94

Table of Contents

1. Introduction

The distribution system is expected to undergo substantial changes with the development of new technologies and the extension of control and communication systems to the lower voltage levels. There are clear benefits which will arise from the integration of protection decisions, control functions and higher level decision-making. An initial project [1] has outlined research directions for applying intelligent system technologies to these protection problems. The areas of protection enhancement that have been considered for development are listed below:

- Adaptive protection.
- Alarm processing.
- High impedance fault detection.
- Fault diagnostics.
- Fault location.
- Restoration.

In [1], the following research and development issues were identified for these problems:

- Sensor placement.
- Robustness under missing or erroneous information.
- Computationally feasible solution techniques.
- Local vs. global computations.
- Software reliability.
- Software flexibility or modularity .
- Software testing and evaluation.
- Analysis of economic impacts.
- Impacts of evolving technology.

In analyzing these issues, it became clear that sensor location is an important first step in the process of enabling the new protection functions. The available on-line information will to a great extent determine the feasible solution techniques. This report details the results of a project for addressing sensor placement schemes with the goal of fault detection and location.

All of the proposed distribution enhancements above are highly dependent on the availability of sensed real-time data. At one extreme, the placement of sensors at every load point could render the computational and logical analysis of faults trivial; however, this would be prohibitively expensive due to the large size of the distribution system. At the other extreme, placement of too few sensors will render some solution approaches completely ineffective due to the lack of information. There have been efforts directed at determining the optimal sensor locations for a narrow range of problems, e.g., state estimation or harmonic source detection. In this work, the objective is to place a few strategic sensors which could effectively aid in computations for two related problems, specifically, short circuit fault location and high impedance fault detection and location. The best sensor placement is also greatly affected by other automation functions including customer interface automation and substation automation.

Due to the diversity of functions needed at the distribution level, it is unlikely that a general solution for sensor placement can be developed (in contrast to state estimation at the transmission level where sensors are placed for observability and redundancy). Rather, the development of different application areas will lead to identification of important variables. These in turn can be combined among the different application areas to identify the appropriate locations for various sensors. The approach taken in this study is to formulate fault location and detection as a state estimation topology problem supplemented with some simple heuristics to improve the analysis. Fault location is achieved based on a combination of sensed quantities and load estimates.

2. Problem Overview

This work focuses on three functions for distribution sensors: short circuit fault location, high impedance fault (HIF) detection and approximate HIF location. In the following, several assumptions are made to define the scope of the problem. These assumptions are:

1. All protective devices are coordinated correctly so that relay and fuse operations do not produce misleading information as to fault location.
2. Status of all switches is known and the circuit structure is radial.
3. No specialized HIF detection equipment (e.g., harmonic analysis to detect arcing) is used.
4. Microprocessor relays are configured so that short circuit fault data is available on-line even if the circuit breaker does not trip.
5. Customer loads can be estimated from recent load flow computations.
6. All customers are consuming power (i.e. no small dispersed generation units) and individual customer data is not metered on-line.

The first two assumptions represent the vast majority of situations and analysis would be greatly complicated without these assumptions. The third assumption is made as the goal of this work is to improve HIF detection and location. Available HIF detection equipment is limited to faults exhibiting arcing and even this functionality is considered by many to be unreliable. The last three assumptions provide the foundation for the work reported here. While together these assumptions may be a bit restrictive, they are useful in representing a typical feeder for which a method of placing a small number of sensors can be evaluated.

2.1 Fault detection

The detection of short circuit faults is straightforward and such faults are assumed to be isolated by protective devices. HIF detection is more difficult. There have been extensive research developments based on harmonic components arising from arcing, e.g. [22], but there remains limitations to the developed techniques. In general, HIFs result in downstream voltage loss for at least one phase and the likely loss of power for one or more customers. This leads to two approaches for detecting HIF: voltage sensing and sensing loss of load. In the simplest sense, detection can be achieved by sensing voltage at all end points of a feeder. The accuracy required for such measurements would be minimal. If one assumes many current sensors are placed on the distribution system, then one can also identify the presence of HIF by loss of load. This loss of load would manifest itself in two ways: one, customers with real-time sensors would be identified as losing power, and two, total load on a feeder would be lower than expected. The latter indication is one that allows the possibility of placing fewer sensors on the system.

2.2 Fault location

Once a fault has been detected, it should be located as accurately as possible so that crews can be effectively dispatched for repair. In radial systems with multiple branches, there may be several possible fault locations which give rise to a specific fault current [1]. If the fault current is known then a set of possible fault locations can be easily found. If the fault is on a lateral protected by a fuse and the fuse has operated correctly, there will be a loss of all customers along that branch. From a circuit viewpoint, the open fuse acts the same as a down-line HIF near the fuse location. Thus, the final step in locating a short circuit fault has much in common with locating a HIF. The HIF location is determined by a combination of sensors which identify out of service customers and state estimation which identifies the loss of load. The next section discusses this state estimation approach.

3. Model Formulation

The state estimation formulation has been extensively studied for the transmission system, e.g. [20]. Recently, there has been interest at the distribution level as well [2-3]. In this section, the problem of fault location is formulated as a state estimation problem where a fault manifests itself as a measurement or topology error.

3.1 State estimation

Weighted Least Squares (WLS) Estimation - A brief review of WLS is presented. More details are widely available, e.g., [7]. A system of measurements $\underline{z}$, a state vector $\underline{x}$ and a measurement function $h(\cdot)$ represent the measurement system.

$$\underline{z} = h(\underline{x}) + \underline{\varepsilon} \tag{1}$$

where $\underline{\varepsilon}$ is a noise or error vector and this system is subject to the constraints:

$$\underline{y} = c(\underline{x}) \tag{2}$$

Typically, the best estimate of $\underline{x}$ is found iteratively based on the linearized equations:

$$\underline{z} = H\underline{x} + \underline{\varepsilon} \tag{3}$$

with

$$\underline{y} = C\underline{x} \tag{4}$$

For simplicity, the linearized formulation will be used throughout without explicitly describing the iterative process. The best estimate of the state can be found by minimizing WLS or WLAV error. Given covariance matrix R for the measurements, the best WLS estimate of the state is:

$$\begin{bmatrix} \underline{\lambda} \\ \underline{\hat{x}}_{wls} \end{bmatrix} = A^{-1} \begin{bmatrix} \underline{y} \\ H^T R^{-1} \underline{z} \end{bmatrix} \tag{5}$$

with

$$A = \begin{bmatrix} 0 & C \\ C^T & H^T R^{-1} H \end{bmatrix}$$

In the WLS formulation, each measurement influences the state estimate based on the degree of confidence in the measurement. A bad measurement is identified by its inconsistency with other measurements. The primary difficulty in implementation of the WLS formulation concerns the rank of A. An A less than full rank leads to unobservable states. One approach to measurement placement focuses on finding zero-pivots in the factorization of A. These zero-pivots indicate unobservable states. In the design phase, unobservability is addressed by additional measurements. During operations, unobservable states require that either a "guess" of the unknown state is made or that observable "islands" are formed. WLS methods have been widely developed for the transmission system and are considered by many to be a mature technology.

Another important problem in WLS estimation concerns the ability to detect measurement errors or the redundancy inherent to the measurement system (see e.g. [6,9,13]). Redundancy is defined as the number of independent measurements minus the number of states *(m-n)*. For an *n* bus system connected to an infinite bus, there are *2n* states. Thus, there must be at least *2n* measurements to obtain a solution (to be observable), plus additional ones for redundancy. The proper placement of these sensors is needed to ensure observability. For example, sensing of all bus injection measurements will ensure observability [14]. Voltage sensor measurements in conjunction with power flow and injection measurements provide additional redundancy.

Weighted Least Absolute Value (WLAV) Estimation - WLAV methods have only recently garnered much interest (see e.g. [7,12,17]) but are growing in importance. WLAV estimation can be formulated as a linear programming problem with the best estimate $\hat{\underline{x}}_{wlav}$ based on weightings $\underline{w}$ as the solution to:

$$\min_{\underline{x}} \underline{w}(\underline{u}+\underline{v}) \tag{6}$$

such that $u_i, v_i \geq 0 \;\; \forall \; i$, and the below hold (eqn 4. repeated here):

$$\underline{y} = C\underline{x} \tag{4}$$

$$\underline{z} = H\underline{x} + \underline{u} - \underline{v} \tag{7}$$

In addition, this formulation can easily incorporate inequality constraints which we'll write by modifying (4) to:

$$\underline{y} \geq C\underline{x} \tag{4a}$$

The WLAV approach is effective in identifying measurement errors. It can be viewed as an approach which selects the subset of the measurements which best represent the system state. A bad measurement will have no influence on the state estimate whereas in WLS estimation every measurement has some influence. In the WLAV approach, an unobservable state may manifest itself as an unbounded solution so that inequality constraint of the form in (4a) may be particularly important. The WLAV approach is less developed and to the authors knowledge there are no utility energy management systems that rely on this method. There are computational concerns with this problem as the number of variable is *'2n+2m'* where *n* is the number of buses (excluding the slack bus) and *m* is the number of measurements. This computational problem may be less of a concern at the distribution level where calculations can be limited to a single feeder or in some cases to a specific lateral.

To summarize the above discussion, let the difference between the actual measurements and the value of the measurements based on the best estimate be defined as the residual vector $\underline{r}$:

$$\underline{r} = \underline{z} - H\underline{\hat{x}} \tag{8}$$

In the WLS formulation, any measurement errors are distributed among different sensors according to the weights and most of the residuals r_i will have small non-zero values. In the WLAV formulation, the measurements that are actually used for the estimate will have zero-valued residuals (excluding round off errors and linearization errors). If $\underline{\varepsilon}$ is Gaussian, the WLS formulation yields the most likely estimate of the state; however, if error conditions exist in either the measurements or the measurement matrix (i.e. topology errors) the WLAV will provide a better estimate and at the same time clearly identify the offending measurement(s).

3.2 Distribution system considerations

Most distribution systems have two characteristics which greatly simplify the state estimation computations: a single generation point and a radial structure. Together these allow one to easily identify any fault conditions by downstream outages. Estimates can be further improved by incorporating the no injection constraint into the best estimate. Location of the voltage sensors is again important. As a general rule, locating the voltage sensors at the far ends of each of the service laterals provides the most information in terms of identifying outages. Since any line section outage causes all downstream customers to lose power, the likelihood of an outage increases as one moves away from the substation. Specifically, the probability of an outage for some customer A is less than or equal to the probability of outages for all customers downstream from A.

The above observation can be incorporated into the WLAV estimate for this problem as follows. Assume that loads along a feeder are forecasted rather than measured. Obviously, the forecast will be invalid for any out of service customers and the likelihood of being out of service increases with distance from the substation. Thus, one can assign decreasing weights to the confidence in the reliability of a forecaste as one moves away from the substation. Specifically:

$$\underline{w}_{wlav} = [w_m, \ldots, w_i, \ldots, 1] \tag{9}$$

with

$$w_i = \sum_{j<i} w_j \tag{10}$$

which will place lower weights as one moves further along the feeder. To avoid numerical problems associated with a wide variation in the weighting values, only points relying on estimates need be weighted as such. The second assumption of a single generation point eliminates the possibility of a customer supplying load to the system. This assumption can be easily incorporated into the constraints (4a) as a fixed positive direction for line flow.

3.3 Proposed approach

Fault Detection: In this work, it is proposed that faults are identified using WLS state estimation. This approach assumes that the faults of concern will appear as bad measurements. The traditional test for a bad measurement is to use a normalized residual test:

$$\sum_{i=1}^{m} r_i^N < \tau \tag{11}$$

where the normalization (superscript N) and threshold τ are based on a statistical analysis (see [21]). An set of measurements is said to contain at least one error when this test fails. For the problem at hand, these residuals may be quite large as many load points will be forecasts rather than true measurements. The normalization process accounts for this fact through inclusion of the standard deviation of forecast errors. Still, large forecasts errors will make the practical application of the proposed technique more difficult.

Fault Location: WLS is not as effective at identifying measurement errors and our tests supported this conclusion. Thus, it is proposed to determine fault location by WLAV state estimation after WLS has detected an error. WLAV will assigns error to as few as measurements as possible. If a large error does occur (which is indicative of a fault), this approach will isolate the offending measurement. If an estimate of the state is desired, the WLS estimation can be repeated with the new topology as the WLAV estimate does not provide the most likely estimate of the state. In general, the accuracy of the fault location is limited by the accuracy of the load estimates.

3.4 Summary: Sensor placement, fault detection and fault location algorithm

Sensor placement

1. Place voltage sensors at all significant feeder end points. Lateral end points can remain unsensed if HIF along the upstream branch are highly unlikely (e.g., line section along a dirt road) or the line section is serving a small load.
2. Sensors should be placed along a lateral to improve the ability to locate faults. Specifically, sense the overall power flow on each lateral with several load points. This can be achieved equivalently by sensing three phase flow of the main feeder between all

branch points (laterals). Laterals can be grouped together by a feeder flow measurement as well.

3. Place further sensors on a lateral so that the expected value of errors in the estimate is less than the load between the sensor and the nearest upstream sensor. These sensors are placed to improve the estimate of the fault location.

Short circuit fault location

1. Faults on main feeder are identified by circuit breaker action (unsuccessful reclosure). Fault can be located by impedance relay data.
2. Fault locations on laterals may be ambiguous; the distance to the fault determining possible fault locations (i.e. those points on laterals equidistant electrically from the substation). The following can clarify location.
    - If available, knowledge of the lateral phase will eliminate some possible fault locations.
    - If the fuse on the lateral has operated correctly then this will appear as a loss of load and electrically identical to a downed conductor HIF at the lateral branch. If any flow or voltage on this lateral is measured, then this can be easily determined. If not then this must be determined by the state estimate.
    - Historical data may be useful if a line section has experienced many faults.

High impedance fault detection and location

1. Faults on main feeder and lateral are detected as a by product of WLS estimation.
2. Fault location is determined by the WLAV approach with the weighting scheme as identified in the previous section. The HIF location is the closest significant residual to the upstream branch.

4. Implementation and Numerical Results

In this section, the implementation of the proposed scheme is discussed. The software has been written in MATLAB. MATLAB was initially a convenient user interface to the FORTRAN LinPack routines. It has evolved into a sophisticated programming framework with an extensive library of computational routines. An actual implementation of the proposed techniques system would probably require lower level code; however, the high level of MATLAB allows very fast prototyping of software. Source code is included in the appendix. This code consists of modules for WLS estimation, WLAV estimation, Newton-Raphson based load flow and other support routines for generating the required matrices. All routines make use of sparse matrices for computational efficiency.

Three types of faults have been considered: HIF through leakage, short-circuit and HIF due to broken conductor. Leakage through a high impedance would be very difficult if not impossible to detect with the method proposed here. Simulation runs confirmed this as only extremely accurate forecasts would allow reliable detection. Further, detection of this type of fault would not be in time to enhance safety. For example, a typical fault of this type may involve heavy machinery that comes into contact with an overhead line. This would require essentially immediate detection to protect the safety of local personnel. For short-circuit faults, it is assumed that the existing overcurrent protection relays operate correctly and the fault is quickly isolated and the short-circuit fault is transformed into an open line fault for the time frame of the state estimation computations. Thus, the rest of this discussion will focus on HIF from down conductor faults. The system measurements during the fault represent a "changed" system and not the system indicated by the measurement function for the unfaulted system. This manifests itself as a significant increase in the residuals, or the error function, of the state estimation. Simulations have shown that faults do produce large residuals and so the above approach is seen to be very effective in detecting open lines. Numerical results for a simple feeder (see Fig. 1) and for a feeder in the PG&E system (the Cotati planning area, see Fig. 2) are described in the following.

Consider the simple feeder of Fig. 1 with several load points. Based on the measurement placement algorithm, flow measurements are placed on the main feeder and on each lateral with several customers. Additionally, a voltage measurement is placed at the end of each lateral and finally, it is assumed there is an estimate for each indicated load. Simulations of high impedance faults were performed throughout the system. Three cases are shown here: down conductor faults in line sections 6-7, 7-8 and 8-9. For each case, errors are introduced into the load estimates to model typical errors in the forecast. These errors are represented by Gaussian (normal) distributed noise. Standard deviation is given as a percentage error for the forecast. In all cases, WLS could easily detect a fault. The results of the WLAV are summarized in Table 1 for 10% noise in load forecasts and in Table 2 for a 20% level of noise in load forecasts. The tabulated values show the measured line flow or forecasted load, the computed flows based on WLAV estimation, the residual error and the difference in percent of the measured or forecasted quantity and the computed value. The most likely fault location is the furthest point upstream with a high percentage error. These values are highlighted in the tables. Notice that in all cases, the faults can be easily located.

The proposed approach has also been applied to a PG&E feeder (Cotati circuit 1102). The area served by this feeder is shown in Fig. 2. The area modeled in this study includes Stony Point Rd., Railroad Ave. and continues on up to East Cotati Ave. As phase information and some line parameters were not known for this feeder, a single phase system with approximated parameters was developed. A one line diagram of the simplified system is shown in Fig. 3 with recommended sensor placement. Table 3 lists the relationship between this diagram and the original feeder. Note that knowledge of the phases was not used in the placement but would enhance the detection process. Voltage sensors are not placed on laterals with only a single load point and current sensors are placed along the feeder to group these laterals. Similar results to the previous example system were obtained for the determination of fault location on this feeder. The results are not shown here for sake of brevity.

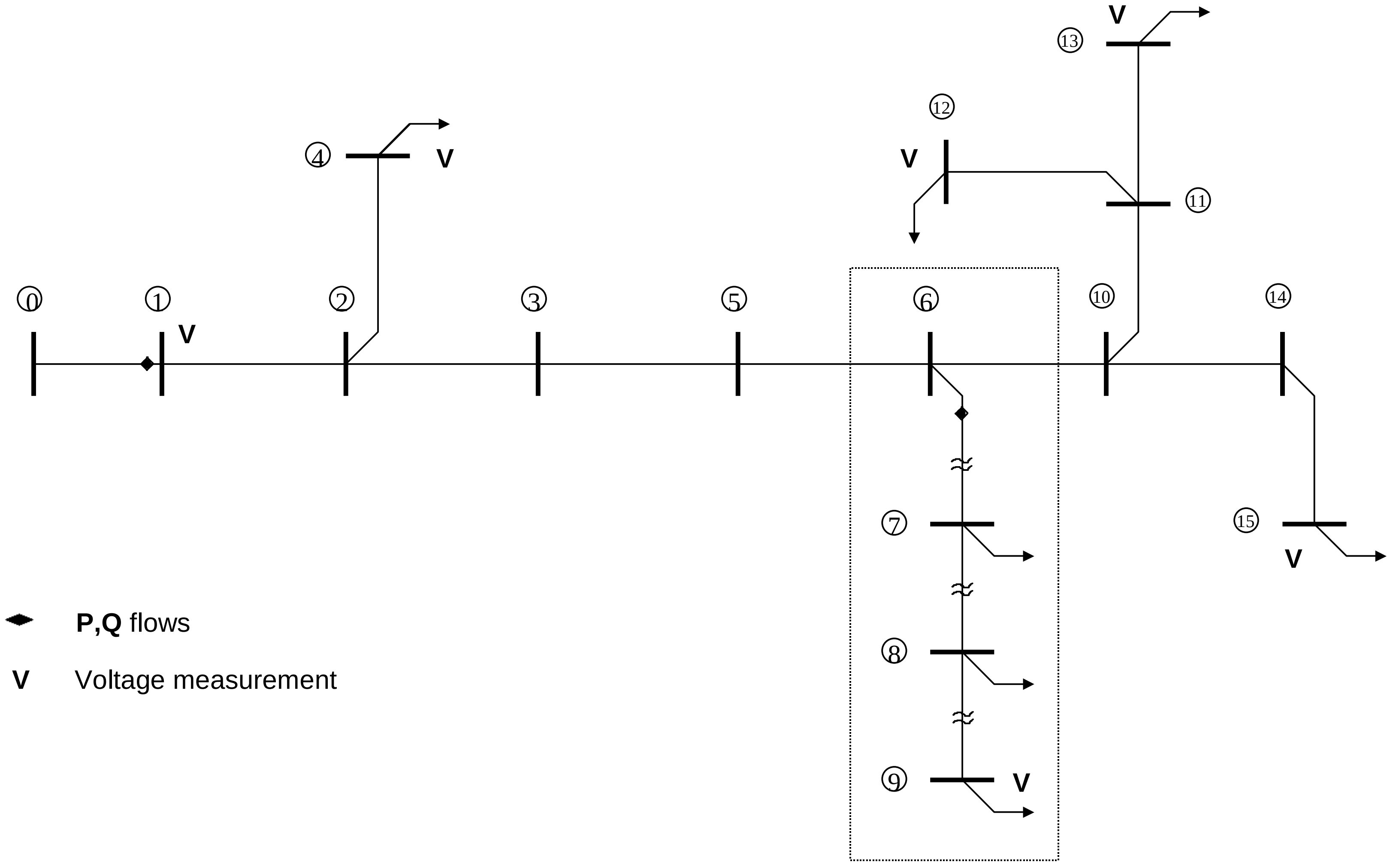
V
⑬
⑫
④
V
V
⑪
⓪
①
②
③
⑤
⑥
⑩
⑭
V
⑦
⑮
V
⑧
⑨
V
◆ **P,Q** flows
V Voltage measurement

Fig. 1 Simple Feeder System

	MEASURED	COMPUTED	RESIDUALS	PERCENT
P-flow 6-7	9.63E-02	9.63E-02	1.86E-13	0%
Q-flow 6-7	4.05E-02	4.05E-02	2.69E-13	0%
P-inject 7	-1.69E-02	-1.69E-02	-4.51E-13	0%
Q-inject 7	-8.11E-03	-8.11E-03	-3.98E-13	0%
P-inject 8	-1.31E-02	-1.31E-02	3.47E-13	0%
Q-inject 8	-4.00E-03	-4.00E-03	1.23E-12	0%
P-inject 9	-7.21E-02	-6.63E-02	-5.80E-03	8%
Q-inject 9	-2.10E-02	-2.84E-02	7.39E-03	35%

NO FAULTS ON SYSTEM

10% NOISE

P-flow 6-7	0.00E+00	0.00E+00	0.00E+00	0%
Q-flow 6-7	0.00E+00	0.00E+00	0.00E+00	0%
P-inject 7	-1.69E-02	-3.24E-14	-1.69E-02	100%
Q-inject 7	-8.11E-03	-5.69E-13	-8.11E-03	100%
P-inject 8	-1.31E-02	-4.97E-13	-1.31E-02	100%
Q-inject 8	-4.00E-03	-2.26E-13	-4.00E-03	100%
P-inject 9	-7.21E-02	3.26E-14	-7.21E-02	100%
Q-inject 9	-2.10E-02	-2.08E-13	-2.10E-02	100%

BRANCH 6 - 7 OPEN

10% NOISE

P-flow 6-7	1.52E-02	1.52E-02	1.22E-13	0%
Q-flow 6-7	1.02E-02	1.02E-02	2.00E-13	0%
P-inject 7	-1.69E-02	-1.52E-02	-1.70E-03	10%
Q-inject 7	-8.11E-03	-8.11E-03	-5.09E-13	0%
P-inject 8	-1.31E-02	-2.32E-13	-1.31E-02	100%
Q-inject 8	-4.00E-03	-2.10E-03	-1.90E-03	48%
P-inject 9	-7.21E-02	7.91E-14	-7.21E-02	100%
Q-inject 9	-2.10E-02	-1.15E-13	-2.10E-02	100%

BRANCH 7 - 8 OPEN

10% NOISE

P-flow 6-7	2.84E-02	2.84E-02	2.22E-13	0%
Q-flow 6-7	1.44E-02	1.44E-02	2.54E-13	0%
P-inject 7	-1.69E-02	-1.69E-02	-9.88E-13	0%
Q-inject 7	-8.11E-03	-8.11E-03	3.11E-14	0%
P-inject 8	-1.31E-02	-1.15E-02	-1.58E-03	12%
Q-inject 8	-4.00E-03	-4.00E-03	5.14E-14	0%
P-inject 9	-7.21E-02	-4.72E-13	-7.21E-02	100%
Q-inject 9	-2.10E-02	-2.30E-03	-1.87E-02	89%

BRANCH 8 - 9 OPEN

10% NOISE

Table 1. Simulation Results on Sample System - 10% Noise in Estimates

	MEASURED	COMPUTED	RESIDUALS	PERCENT
P-flow 6-7	7.29E-02	7.29E-02	-1.29E-13	0%
Q-flow 6-7	3.12E-02	3.12E-02	-1.89E-13	0%
P-inject 7	-1.48E-02	-1.48E-02	5.98E-14	0%
Q-inject 7	-8.37E-03	-8.37E-03	2.07E-13	0%
P-inject 8	-1.48E-02	-1.48E-02	2.97E-13	0%
Q-inject 8	-2.44E-03	-2.44E-03	2.42E-13	0%
P-inject 9	-6.62E-02	-4.33E-02	-2.29E-02	35%
Q-inject 9	-1.47E-02	-2.04E-02	5.68E-03	39%

NO FAULTS ON SYSTEM

20% NOISE

P-flow 6-7	0.00E+00	0.00E+00	0.00E+00	0%
Q-flow 6-7	0.00E+00	0.00E+00	0.00E+00	0%
P-inject 7	-1.48E-02	-6.41E-14	-1.48E-02	100%
Q-inject 7	-8.37E-03	-5.26E-13	-8.37E-03	100%
P-inject 8	-1.48E-02	-4.97E-13	-1.48E-02	100%
Q-inject 8	-2.44E-03	-2.26E-13	-2.44E-03	100%
P-inject 9	-6.62E-02	3.26E-14	-6.62E-02	100%
Q-inject 9	-1.47E-02	-2.08E-13	-1.47E-02	100%

BRANCH 6 - 7 OPEN

20% NOISE

P-flow 6-7	1.15E-02	1.15E-02	-2.07E-13	0%
Q-flow 6-7	7.87E-03	7.87E-03	-2.70E-13	0%
P-inject 7	-1.48E-02	-1.15E-02	-3.27E-03	22%
Q-inject 7	-8.37E-03	-7.87E-03	-4.95E-04	6%
P-inject 8	-1.48E-02	2.89E-13	-1.48E-02	100%
Q-inject 8	-2.44E-03	-6.13E-13	-2.44E-03	100%
P-inject 9	-6.62E-02	1.44E-13	-6.62E-02	100%
Q-inject 9	-1.47E-02	1.49E-13	-1.47E-02	100%

BRANCH 7 - 8 OPEN

20% NOISE

P-flow 6-7	2.15E-02	2.15E-02	-2.09E-14	0%
Q-flow 6-7	1.11E-02	1.11E-02	-6.12E-15	0%
P-inject 7	-1.48E-02	-1.48E-02	-9.88E-14	0%
Q-inject 7	-8.37E-03	-8.37E-03	-1.37E-13	0%
P-inject 8	-1.48E-02	-6.70E-03	-8.11E-03	55%
Q-inject 8	-2.44E-03	-2.44E-03	1.68E-13	0%
P-inject 9	-6.62E-02	1.28E-13	-6.62E-02	100%
Q-inject 9	-1.47E-02	-2.96E-04	-1.44E-02	98%

BRANCH 8 - 9 OPEN

20% NOISE

Table 2: Simulation Results on Simple System - 20% Noise in Estimates

Fig. 2. Cotati Feeder Area (PG&E Drawing)

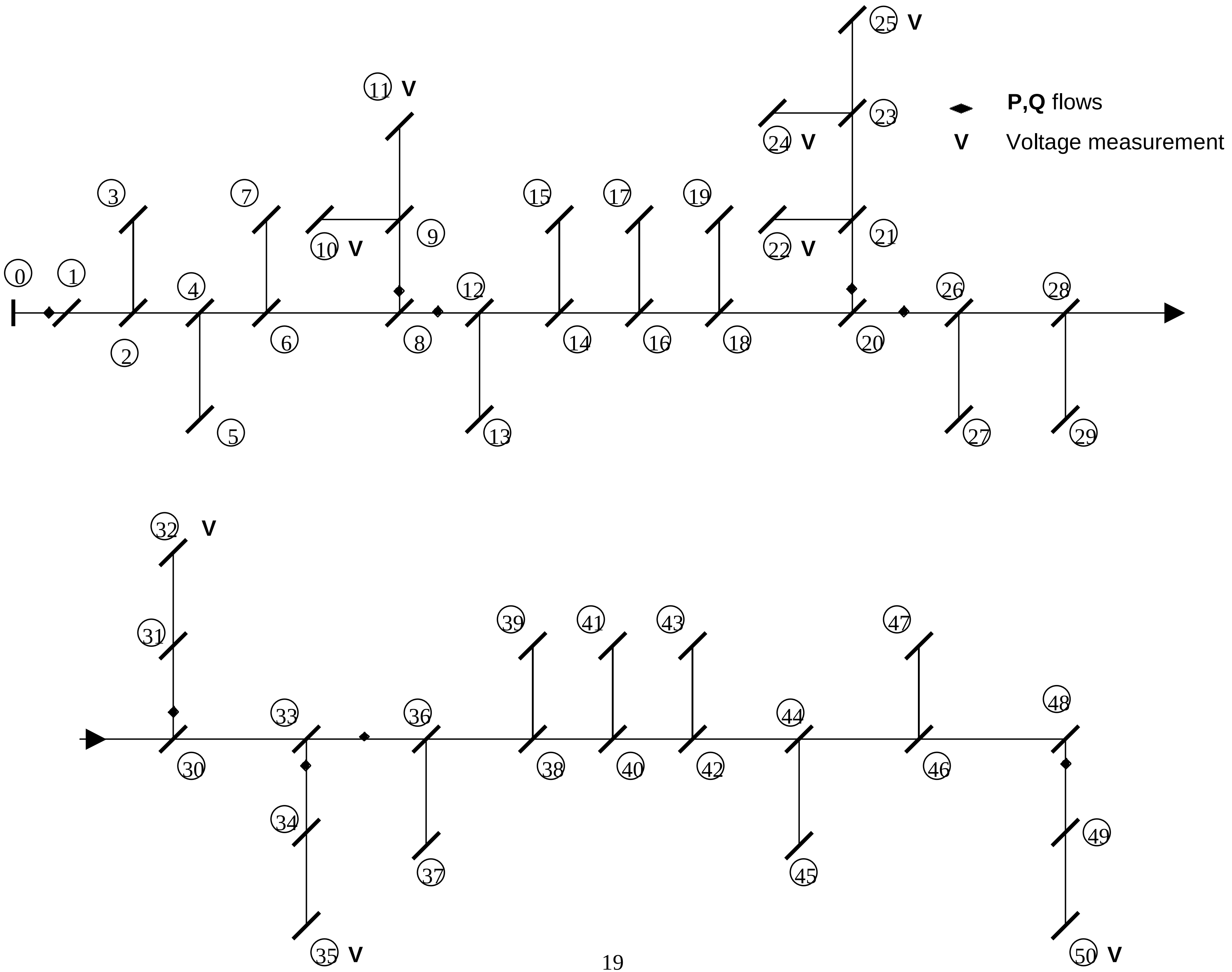
P,Q flows
V Voltage measurement
0
1
2
3
4
5
6
7
8
9
10 V
11 V
12
13
14
15
16
17
18
19
20
21
22 V
23
24 V
25 V
26
27
28
29
30
31
32 V
33
34
35 V
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50 V

Fig. 3. Cotati Circuit 1102

Table 3: Feeder Data for Cotati Circuit 1102

5. Discussion and Conclusions

This report has outlined a general approach for placing sensors to detect and locate short-circuit faults and HIF on the distribution system. The approach is formulated first as a state estimation problem where a fault appears similar to a failed measurement. The state estimation analysis needs to be supplemented with some simple logical analysis. Numerical examples have shown the feasibility for this proposed approach. A full development would require the following steps.

1. Investigation of the accuracy to be expected from load estimates and the resulting ability to locate faults.

2. Investigation of the computational concerns of the proposed method for distribution systems concurrent with the reworking of the code developed here into a more efficient language such as C or FORTRAN.

3. Appropriate microprocessor-based protection settings to yield fault data for short-circuit faults on laterals protected by fuses.

4. Consideration of the cost/benefits for coverage of fault detection and location. For example, some customers may not require or benefit significantly from the ability to identify HIF.

5. This work has not considered coordination with other applications which may require sensors, e.g., load management.

References

[1] K. Tomsovic, R. Stephenson, F. Soudi and J. Yee, "Intelligent Technologies for Distribution Protection," *PG&E Internal Report*, July 1994.

[2] Baran, M.E. and Kelley, A. W., “State Estimation for Real-Time Monitoring of Distribution Systems.” *IEEE Transactions on Power Systems*, Vol. 9, No. 3, August 1994, pp. 1601-1609.

[3] Lu, C.N., Teng, J.H. and Liu, W.H.E., “Distribution System State Estimation.” *IEEE/PES 1994 Winter Meeting*, paper no. 94 WM 098-4 PWRS.

[4] Chen, T. et. al., "Distribution System Power Flow Analysis - A Rigid Approach." *IEEE Transactions on Power Delivery*, Vol. 6, No. 3, July 1991, pp. 1146-1152.

[5] Halpin, S.M. et. al., "An Improved Fault Analysis Algorithm For Unbalanced Multi-Phase Power Distribution Systems." IEEE Transactions on Power Delivery, Vol. 9, No. 3, July 1994, pp. 1332-1338.

[6] Mili, L., Cutsem, T.V. and Ribbens-Pavella, M., "Bad Data Identification Methods in Power System State Estimation - A Comparative Study." *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-104, No. 11, November 1985, pp. 3037-3049.

[7] Kotiuga, W.W. and Vidyasagar, M., "Bad Data Rejection Properties of Weighted Least Absolute Value Techniques Applied to Static State Estimation." *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-101, No. 4, April 1982, pp. 844-851.

[8] Holten, L., Sverre Aam, A.G. and Wu, F.F., "Comparison of Different Methods for State Estimation." *IEEE Transactions on Power Systems*, Vol. 3, No. 4, November 1988, pp. 1798-1806.

[9] Handschin, E., Schweppe, F.C., Kohlas, J. and Fliechter, A., "Bad Data Analysis for State Estimation." *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-94, No. 2, March/April 1975, pp. 329-337.

[10] Krumpoltz, G.R., Clements, K.A. and Davis, P.W., "Power System Observability: A Practical Algorithm using Network Topology." *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-99, No. 4, July/August 1980, pp. 1534-1542.

[11] Baldwin, T.L., Mili, L., Boisen, M.B.,Jr. and Adapa, R., "Power System Obsevability with Minimal Phasor Measurement Placement." *IEEE Transactions on Power Systems*, Vol. 8, No. 2, May 1993, pp. 707-715.

[12] Abur, A. and Celik, M.K., "Least Absolute Value State Estimation with Equality and Inequality Constraints." *IEEE/PES 1994 Summer Meeting*, paper no. 92 SM 581-9 PWRS.

[13] Koglin, H.J., Neisius, T., Beissler, G. and Schmitt, K.D., "Bad Data Detection and Identification." *International Journal of Electrical Power and Energy Systems*, Vol. 12 Number 2, April 1990, pp. 94-103.

[14] Clements, K.A., "Observability Methods and Optimal Meter Placement." *International Journal of Electrical Power and Energy Systems*, Vol. 12 Number 2, April 1990, pp. 88-93.

[15] Clements, K.A., Krumpholz, G.R., and Davis, P.W., “State Estimator Measurement System Reliability Evaluation - An Efficient Algorithm Based on Topological Observability Theory.” *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-101, No. 4, April 1982, pp. 997-1004.

[16] Korres, G.N. and Contaxis, G.C., “Identification and Updating of Minimally Dependent Sets of Measurements in State Estimation.” *IEEE Transactions on Power Systems*, Vol. 6, No. 3, August 1991, pp. 999-1005.

[17] Abur, A. and Celik, M.K., “A Fast Algorithm for the Weighted Least Absolute Value State Estimation.” *IEEE Transactions on Power Systems*, Vol. 6, No. 1, February 1991, pp. 1-8.

[18] Irving, M.R., Owen, R.C. and Sterling, M.J.H., “Power System State Estimation Using Linear Programming.” *Proceedings of the IEE*, Vol. 125, No. 9, September 1978, pp. 879-885.

[19] Clements, K.A., Woodzell, G.W. and Burchett R.C., “A New Method for Solving Equality-Constrained Power System Static-State Estimation.” *IEEE Transactions on Power Systems*, Vol. 5, No. 4, November 1990, pp. 1260-1265.

[20] Wu, F.F., “Power System State Estimation: A Survey.” *International Journal of Electrical Power and Energy Systems*, Vol. 12, Number 2, April 1990, pp. 80-87.

[21] Grainger, J.J., Stevenson, W.D.,Jr., *Elements of Power System Analysis*, McGraw-Hill, 1994.

[22] *Detection of High Impedance Faults*, EPRI Report EL-2413, June 1982.

Appendix A: MATLAB Code

Note:

(1) Lines beginning with % indicate comment lines. All other lines represent code.
(2) All code uses standard MATLAB routines.

A.1 Weighted Least Squares Estimation

```
function [x,y,v,J,resid] = wlse(n,zfile,hfile,wfile,z,x0);
%   [x,y,v,J,resid] = wlse(n,zfile,hfile,wfile,z,x0);
%
%   ‘x’ is the system state
%   ‘y’ is ybus matrix
%   ‘v’ is the system voltages
%   ‘J’ is the performance index
%   ‘resid’ is the measurement residuals
%   'n' is the number of system buses.
%   'zfile' is the system z parameters.
%   'hfile' is the coded (on/off) bus voltage measurements.
%   'wfile' is the measurement weight parameters.
%   'z'  is a column vector of the measurements in volts, watts and vars:
%          Z1 = voltage magnitude at bus x
%          .            .
%          .            .
%          Zn-1 = injected real power at bus n
%          Zn = injected reactive power at bus n
%   ‘x0’ initial estimate including slack bus
%   'J' is the returned weighted sum of the measurement errors squared.

%build ybus
y = sparse([],[],[],n+1,n+1,0);
  if zfile(1) == 0
     error('zfile(1) cannot equal 0')
  end
y(1,2)= -inv(zfile(1));
zcnt = 2;
  for ai = 2:n+1
    for aj = ai:n+1
      if zfile(zcnt) ~= 0
```

```
           y(ai,aj) = -inv(zfile(zcnt));
      end
      zcnt = zcnt + 1;
    end
  end
yd = diag(y);
y = y - diag(yd);
y = y + y.';
y = y - diag(sum(y)) - diag(yd);
y = y*12.47^2;    %base admittance = 12.47^2;

% initial x-vector
x = zeros(2*n,1);
x(1:n) = ones(n,1);

% create top part of hx matrix (linearized measurement matrix)
ubj = length(hfile);
hu = sparse([],[],[],ubj,2*n,0);
  for cnt = 1:ubj
    hu(cnt,hfile(cnt)) = 1;
  end

% weight matrix definition
d = 1:length(wfile);
w = sparse(d,d,wfile);

% Main iteration loop
infnorm = 1;  iter = 0;
while infnorm > 1e-12

  % jacobian calculation
  step = 0.000001;
  hlow = wlshfn_b(n,x,y,x0);
  hl = sparse([],[],[],2*n+2,2*n,0);
    for cnt = 1:2*n
       tempx = x;
       tempx(cnt) = x(cnt) + step;
       temp1 = wlshfn_b(n,tempx,y,x0);
       hl(:,cnt) = (temp1 - hlow)/step;
    end

% Completion of measurement matrix
```

```
  hx = [hu;hl];
  hup = zeros(ubj,1);
    for cnt = 1:ubj
      hup(cnt,1) = x(hfile(cnt));
    end
  h =  [hup;hlow];

% error
  e = z - h;
  G = full(hx'*w*hx);

% updates
 deltax = inv(G)*hx'*w*e;
 x = x + deltax;

  infnorm = norm(deltax,inf);
  iter = iter + 1;
  if iter > 14
    fprintf('CAUTION - WLS has not converged in 15 iterations.\r\n')
    fprintf('convergance criteria: 1e-12, infinity norm: %g\r\n\r\n', infnorm)
    break
  end
end

% Final residual calculation
v = [ x(1:n), x(n+1:2*n)];
hup = zeros(ubj,1);
  for cnt = 1:ubj
    hup(cnt,1) = x(hfile(cnt));
  end
h= [hup;wlshfn_b(n,x,y,x0)];  %  Construction of measurement matrix (See A.3).
ee = z - h;
J = sum(w * ee .* ee);
resid = ee;
```

A.2 Weighted Least Absolute Value Estimation

```
function [x,y,v,J,resid] = wlavei(n,zfile,hfile,wfile,tfile,z,x0);
%   [x,y,v,J,resid] = wlavei(n,zfile,hfile,wfile,tfile,z);
%
% Weighted Least Absolute Value with Equality and Inequality constraints
%   ‘x’, ‘y’, ‘v’, ‘J’, ‘resid’ as before
%   'n' is the number of system buses.
%   'zfile' is the system z parameters.
%   'hfile' is the coded (on/off) bus voltage measurements.
%   'wfile' is the measurement weight parameters.
%   ‘tfile’ is a matrix of tolerances for inequalities
%   'z'  is a column vector of the measurements in volts, watts and vars:
%         Z1 = voltage magnitude at bus x
%         .            .
%         .            .
%         Zn-1 = injected real power at bus n
%         Zn = injected reactive power at bus n
%   ‘x0’ initial estimate including slack bus

%build ybus
y = sparse([],[],[],n+1,n+1,0);
  if zfile(1) == 0
     error('zfile(1) cannot equal 0')
  end
y(1,2)= -inv(zfile(1));
zcnt = 2;
  for ai = 2:n+1
    for aj = ai:n+1
      if zfile(zcnt) ~= 0
           y(ai,aj) = -inv(zfile(zcnt));
      end
      zcnt = zcnt + 1;
    end
  end
yd = diag(y);
y = y - diag(yd);
y = y + y.';
y = y - diag(sum(y)) - diag(yd);
y = y*12.47^2;    %base admittance = 12.47^2;
```

```
% initial x-vector
x = zeros(2*n,1);
x(1:n) = ones(n,1);


% create top part of hx matrix
ubj = length(hfile);
hu = sparse([],[],[],ubj,2*n,0);
  for cnt = 1:ubj
    hu(cnt,hfile(cnt)) = 1;
  end

% number of measurements
  m = length(z);

% Main iteration loop
infnorm = 1;  iter = 0;
while infnorm > 1e-12
  % jacobian calculation
  step = 0.000001;
  hlow = wlshfn(n,x,y);
   hl = sparse([],[],[],2*n+2,2*n,0);
    for cnt = 1:2*n
       tempx = x;
       tempx(cnt) = x(cnt) + step;
       temp1 = wlshfn(n,tempx,y);
       hl(:,cnt) = (temp1 - hlow)/step;
    end
  hx = [hu;hl];
  hx = full(hx);

  % h=h(x) calculation
  hup = zeros(ubj,1);
    for cnt = 1:ubj
      hup(cnt,1) = x(hfile(cnt));
    end
  h =  [hup;hlow];

% Construction of constraint matrix A
Ae = hx;
  A_lte =  hx;
  A_gte = -hx;
```

```
Za = z - h;
Ze = z - h;
Z_lte =  tfile(:,2) - h;
Z_gte = -tfile(:,1) + h;

for cnt = m:-1:1
 if tfile(cnt,1) == 0
  if tfile(cnt,2) == 0
   if z(cnt) == 0       % tfile(cnt,1)=0, tfile(cnt,2)=0, z=0
    A_lte(cnt,:) = [];
    A_gte(cnt,:) = [];
    Z_lte(cnt) = [];
    Z_gte(cnt) = [];
   else             % tfile(cnt,1)=0, tfile(cnt,2)=0
    Ae(cnt,:) = [];
    A_lte(cnt,:) = [];
    A_gte(cnt,:) = [];
    Ze(cnt) = [];
    Z_lte(cnt) = [];
    Z_gte(cnt) = [];
   end
  else           % tfile(cnt,1)=0, tfile(cnt,2)~=0
   Ae(cnt,:) = [];
   A_gte(cnt,:) = [];
   Ze(cnt) = [];
   Z_gte(cnt) = [];
  end
 else             % tfile(cnt,1)~=0, tfile(cnt,2)=0
  if tfile(cnt,2) == 0
   Ae(cnt,:) = [];
   A_lte(cnt,:) = [];
   Ze(cnt) = [];
   Z_lte(cnt) = [];
  else          % tfile(cnt,1)~=0, tfile(cnt,2)~=0
   Ae(cnt,:) = [];
   Ze(cnt) = [];
  end
 end
end
f = [zeros(2*n,1); wfile; wfile];
dim_Ae = size(Ae,1);
dim_A_lte = size(A_lte,1);
```

```
  dim_A_gte = size(A_gte,1);
  A=[hx, eye(m), -eye(m);...
     Ae, zeros(dim_Ae, 2*m);...
     A_lte, zeros(dim_A_lte, 2*m);...
     A_gte, zeros(dim_A_gte, 2*m);...
     zeros(2*m ,2*n), -eye(2*m)];

% constraint vector and dimensioning
  b = [Za; Ze; Z_lte; Z_gte; zeros(2*m,1)];
  N = m + dim_Ae;

  %  X=LP(f,A,b,VLB,VUB,X0,N) solves the linear programming problem:
  %
  %      min f'x    subject to:   Ax <= b
  %       x
  %
  % VLB,VUB define a set of lower and upper bounds on the design
  % variables, X, so that the solution is always in the
  % range VLB < X < VUB.  X0 sets the initial starting point to X0.
  % N indicates that the first N constraints defined by A and b
  % are equality constraints.

X=lp(f,A,b,[],[],[],N);

  x = x + X(1:2*n);
  deltax = X(1:2*n);
  infnorm = norm(deltax,inf);
  iter = iter + 1;
  if iter > 14
    fprintf('CAUTION - WLAVEI has not converged in 15 iterations.\r\n')
    fprintf('convergance criteria: 1e-12, infinity norm: %g\r\n\r\n', infnorm)
    break
  end
end
v = [ x(1:n), x(n+1:2*n)];

% Final residue calculations
hup = zeros(ubj,1);
  for cnt = 1:ubj
    hup(cnt,1) = x(hfile(cnt));
  end
h = [hup;wlshfn_b(n,x,y,x0)];
```

```
resid = z - h;
J = wfile'*abs(resid);
```

A.3 Supplementary Routines

Construction of measurment matrix

```
function h = wlshfn(n,x,y,x0)
% function h = wlshfn(n,x,y,x0)
%
%   ‘x’, ‘y’, ‘x0’ as before
% 'h' is a column vector of the measurement functions.
%      i.e.:     P from bus 0 to bus 1
%                Q from bus 0 to bus 1
%                P1
%                Q1
%                ...
%                Pn
%                Qn

v = zeros(n+1,1);
v(1) = x0(1) .* exp(i*x0(2));
v(2:n+1,1) = x(1:n) .* exp(i*x(n+1:2*n));
s = conj(y)*conj(v).*v;
h = zeros(2*n+2,1);
h(1,1) = real(conj(-(v(1)-v(2))*y(1,2))*v(1));
h(2,1) = imag(conj(-(v(1)-v(2))*y(1,2))*v(1));
 for cnt = 2:n+1
   h(2*cnt-1,1) = real(s(cnt));
   h(2*cnt,1)  =  imag(s(cnt));
 end
```

Newton-Raphson Load Flow

```
function [y,x,s1,v] = nr(n,zfile,s)
%  [y,x,s1,v] = nr(n, zfile, s)
%
%   ‘x’, ‘y’, ‘zfile’ as before
%   'n' is the number of system buses.
%   's1' is the power from bus 0 to bus 1.
%   's' is a column vector containing the injected real and reactive powers in watts and
%         vars.      (Base MVA is 1e6.)
```

```
%           format: s(1) = real injected power at bus 1
%                   s(2) = reactive injected power at bus 1
%                   .
%                   .
%                   s(2n-1) = real injected power at bus n
%                   s(2n) = reactive injected power at bus n

%build ybus
y = sparse([],[],[],n+1,n+1,0);
  if zfile(1) == 0
     error('zfile(1) cannot equal 0')
  end
y(1,2)= -inv(zfile(1));
zcnt = 2;
  for ai = 2:n+1
    for aj = ai:n+1
      if zfile(zcnt) ~= 0
          y(ai,aj) = -inv(zfile(zcnt));
      end
      zcnt = zcnt + 1;
    end
  end
yd = diag(y);
y = y - diag(yd);
y = y + y.';
y = y - diag(sum(y)) - diag(yd);
y = y*12.47^2;    %base admittance = 12.47^2;

% initial x-vector
x = zeros(2*n,1);
x(1:n) = ones(n,1);

% Main iteration loop
infnorm = 1; iter = 0;
while infnorm > 1e-12
  % jacobian calculation
  hx = zeros(2*n);
  step = 0.000001;
  h = nrhfn(n,x,y);
  h = h(3:2*n+2);
    for cnt = 1:2*n
       tempx = x;
```

```
      tempx(cnt) = x(cnt) + step;
      temp1 = nrhfn(n,tempx,y);
      temp1 = temp1(3:2*n+2);
      hx(:,cnt) = (temp1 - h)/step;
    end

  e = s - h;
  deltax = inv(hx)*e;
  x = x + deltax;
  infnorm = norm(deltax,inf);
  iter = iter + 1;
  if iter > 14
    fprintf('CAUTION - NR has not converged in 15 iterations.\r\n')
    fprintf('convergance criteria: 1e-12, infinity norm: %g\r\n\r\n', infnorm)
    break;
  end
end

v = [ x(1:n), x(n+1:2*n)];

v1 = x(1)*exp(i*x(n+1));
i1 = -(1 - v1)*y(1,2);
ss = v1*conj(i1);
s1 = [real(ss); imag(ss)];
```

Subrroutine to assist for Newton-Raphson Load Flow

```
function h = nrhfn(n,x,y)
% function h = nrhfn(n,x,y)
%
% ‘n’, ‘x’, ‘y’ as before
% 'h' is a column vector of the injected system real and reactive powers.
%     i.e.:    P1
%              Q1
%              ...
%              Pn
%              Qn

v = zeros(n+1,1);
v(1) = 1;
v(2:n+1,1) = x(1:n) .* exp(i*x(n+1:2*n));
```

```
s =conj(y)*conj(v).*v;

h = zeros(2*n+2,1);
 for cnt = 1:n+1
   h(2*cnt-1,1) = real(s(cnt));
   h(2*cnt,1)  =  imag(s(cnt));
 end
```