Fault Location Using Sparse IED Recordings

M. Kezunovic, *Fellow, IEEE, and* M. Knezev, *Student Member, IEEE*

***Abstract*--Basic goal of power system is to continuously provide electrical energy to users. Like with any other system, failures in power system can occur. In those situations it is critical that remedial actions are applied as soon as possible. To apply correct remedial actions it is very important that accurate fault condition and location are detected. In this paper, different fault location algorithms followed with description of intelligent techniques used for implementation of corresponding algorithms are presented. New approach for fault location using sparse measurements is examined. According to available data, it decides between different algorithms and selects an optimal one. New approach is developed by utilizing different data structures in order to efficiently implement algorithm decision engine, which is presented in paper.**

***Index Terms*—fault location, genetic algorithms, neural networks, power system monitoring, substation measurements, sampling synchronization, tree data structures**

I. Nomenclature

CBM - Circuit Breaker Monitor
DFR - Digital Fault Recorder
DPR – Digital Protection Relay
FL - Fault Location
GA – Genetic Algorithm
IEDs - Intelligent Electronic Devices
NN- Neural Network
OPFL - Optimized Fault Location
SCADA - Supervisory Control and Data Acquisition
XML – Extensible Markup Language

II. Introduction

Various types of faults appear in power system. Faults can appear due bad whether conditions, equipment damage, equipment failure, environment changes and many other reasons. Once fault appears, remedial actions are immediately taken by protection equipment. In parallel, different IEDs will automatically notice the fault as abnormality and record corresponding current, voltage and status signals, which are used for fault investigation. They can be done automatically if fault location algorithm is implemented in the IED or manually by the user that investigates fault. Usually users that investigate fault have to wait before any analysis is done for data to be retrieved from different substations, which can last up to few hours or even days in some cases.

In general most IEDs immediately calculate FL using one-end FL algorithms if they have measurements available only from one side of line [1]. In case when devices are able to communicate between each other, they are capable to use two-end FL algorithm. Two-end FL algorithm is more accurate then one-end FL algorithm, but it is not so common that communication channel between two IEDs exists. Sometimes there are no recordings available close to a fault. For a case shown on Fig. 1, depending on fault, various DFRs are triggered. All of them may be distant to fault location.

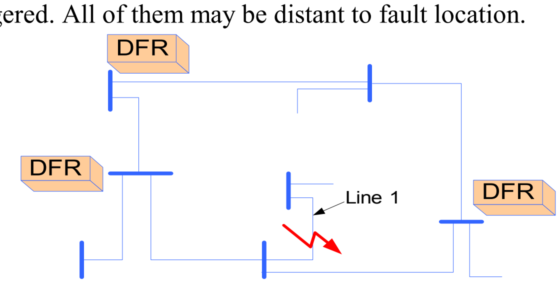


Fig. 1. Layout of closest DFRs to a fault in case of fault present on Line1.

We can recognize two goals that should be achieved:

a) Speeding up fault location procedure,
b) Applying the most suitable algorithm on available recordings that correspond to same fault event.

In [2] authors propose solution in which recordings from different IEDs are automatically transferred to central repository. In rest of the paper we will assume that such repository of DFR recordings is available to us. Depending on availability of data and network parameters, different algorithms are more or less suitable for calculating FL. We propose solution that is capable of selecting optimal fault location algorithm based on available recordings and topology information.

First, architecture of proposed solution is presented. It shows relations between input data and corresponding algorithms. Then, synchronized sampling two-end FL algorithm is described and usage of NN for classifying faults and selecting a section where the fault may be is presented. Similarly system-wide sparse measurement algorithm and use of GA in this algorithm will be explained. One-end FL will be briefly presented in order to clarify influence of this algorithm. Finally, evaluation of the use of above algorithms is summarized in OPFL algorithm. This algorithm is implemented using decision tree, which is described in detail.

This work was supported in part by Concurrent Technologies Corporation and NSF I/UCRC Power Engineering Research enters.

M. Kezunovic and M. Knezev are with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77840 USA (e-mails: kezunov@ece.tamu.edu, mknezev@tamu.edu).

III. Architecture of Proposed Solution

Since proposed solution should be able to use different FL algorithms it is necessary to provide different external tools in order to achieve the optimal performance of each algorithm. Architecture of the solution is shown on Fig. 2. Two segments can be recognized:

1) Fault Location Module
2) External Tools Module

FL module updates power system status with retrieved data, process new event files and runs the most suitable fault location algorithm. Final results are exported into fault report. FL algorithms that are used as a possible selection include:

a) Synchronized sampling two-ended algorithm [3]
b) Unsynchronized sampling two-end algorithm [4]
c) System-wide sparse measurement algorithm [5]
d) Phasor-based single ended algorithm [6]
e) Single ended FL using symmetrical components [1]

Optimal FL algorithm selection is done by OPFL algorithm. Each of these algorithms will be presented in more details later.

External tools module consists of:

a) SCADA EMS PI Historian used for obtaining the latest load, branch and generator data in order to update system model before FL calculation starts.
b) DFR Assistant [7] provides new event recordings from central repository in COMTRADE format [8] and preliminary fault report. Report describes behavior of protection equipment, recognizes type of fault and it is used by other algorithms as input file.
c) PSS/E Short Circuit program [9] is accessed during fault calculation by some algorithms in order to run power flow and short circuit analysis automatically.
d) System model in PSS/E format is updated before any calculation starts in order to reflect system state prior to a fault. This is very important feature especially if topological changes took place in the mean time.

It can be noticed from Fig. 2 that proposed solution has modular architecture, which enables expanding this solution with additional segments. As shown in [2], recordings from different IEDs (DFR, DPR, and CBM) are available in central repository. Processing additional data collected from these devices would provide more information about protection equipment that has operated. This information could be very useful for reducing possible fault location area; understanding nature of fault and it could be applied as input parameter for FL algorithms.

IV. Synchronized Sampling Two-Ended Algorithm

Today, hardware and software tools needed for obtaining data from two ends are becoming much more accessible. Installing such architecture became very rewarding especially for critical transmission lines. In [3] FL using synchronized sampling at two ends of a transmission line is presented. This algorithm has nice property that it doesn't depend on any setting, which makes it very robust, and results are very accurate (obtained error is 0.5% in most cases [3]). This method is used as off-line tool for calculating FL. If used as the protective relaying application, it must be executed in real-time. Because of this, a trade-off between the accuracy and speed of decision is made. In order to improve both relaying (real-time) and fault location (off-line) decision-making, in [10] authors propose enhancing application of synchronized sampling two-end FL algorithm, by introducing technique for fault detection and classification based on a specific neural network.

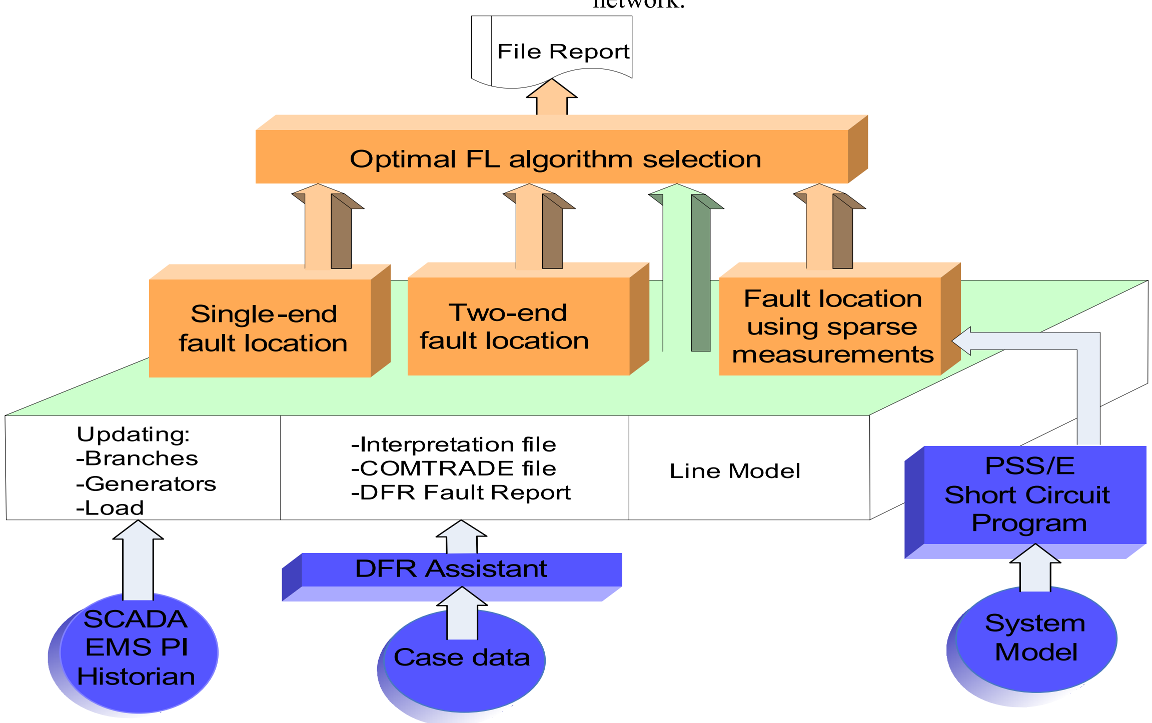


Fig. 2. Architecture of proposed solution

A. *Description of Algorithm*

Synchronized sampling at two ends of a transmission line belongs to time based methods and it uses both lumped and distributed model depending on transmission line length. This algorithm is based on fact that the voltages and currents from one end of the faulted line can be expressed in term of the voltages and currents of the opposite end. So in case that fault occurs at some point x on transmission line as Fig. 3 shows we have [3]:

$$v_F = L^v\{v_S, i_S, d-x\} \tag{1}$$

$$v_F = L^v\{v_R, i_R, x\} \tag{2}$$

By combining (1) and (2) we get:

$$L^v\{v_S, i_S, d-x\} - L^v\{v_R, i_R, x\} = 0 \tag{3}$$

Where v_S, i_S, v_R, i_R are vectors of voltages and currents of sending and receiving end and L^v is operator that defines mathematical model of the line. Properties of this algorithm will be discussed more in section VII.

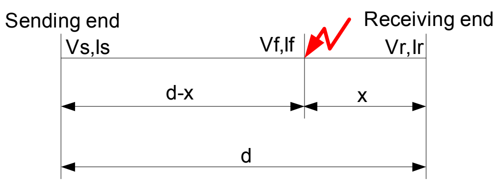


Fig. 3. Faulted transmission line

B. *Implementation and usage of neural networks*

In order to satisfy protective relaying requirements fault must be detected and classified in real-time, but the calculations still has to be robust and reliable technique must be used. Using NN algorithm for fault detection and classification as well as fault location (section determination) is one solution. Unlike the common algorithms NN usage enables independence from commonly varying parameters: type of fault, fault location, fault impedance, voltage levels etc. NNs are trained by individual examples to capture general, always complex and nonlinear, relationships among data.

Extracting system behavior into concise representation from large data sets can be completed by using NNs. This procedure is called clustering. The idea is to recognize patterns among given data and sort them into different clusters. All patterns that belong to the same cluster should be as similar as possible; on the other hand patterns that belong to different clusters should be as different as possible [10]. In case that we directly apply samples of voltages and currents measured from one end of line as input data, by matching corresponding patterns it is possible to classify input data into different clusters as Fig. 4 shows. Clustering of input data may be done by using self-organizing maps as special types of NNs. Procedure is described in detail in [11].

In the case of protective relaying application there are different states of faulted line (Normal, AG, BG, CG, AB/ABG, BC/BCG, AC/ACG, ABC/ABCG, Zone I, Zone II etc.). After training, different clusters are established as Fig. 5 shows.

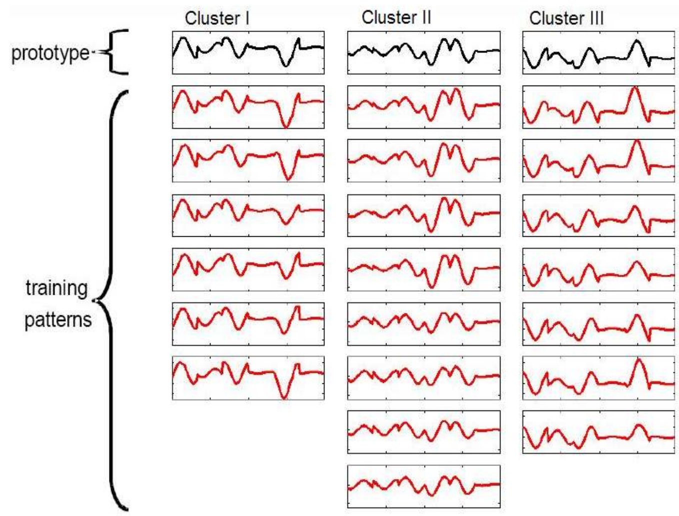


Fig. 4. Classifying clusters according to similar patterns [10]

When new set of input signals is available, appropriate conditioning of input data is done in order to make them comparable with prototype cases. If the input pattern is in "normal-state", data window is shifted for one sample and comparison is repeated again. In case that pattern belongs to faulted case, further classification is done. At the end, the state is classified as belonging to some of the existing clusters.

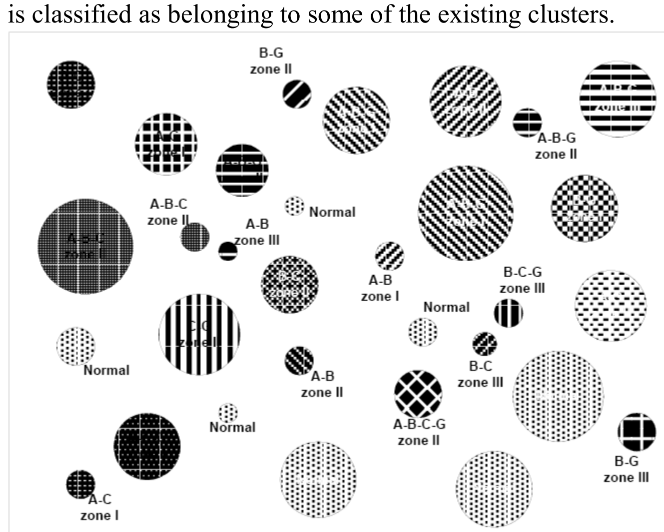


Fig. 5. Established data structures after training [11]

V. System-Wide Sparse Measurement Algorithm

Although there are many accurate two-end, three-end algorithms, they are not always applicable because only data from limited number of substation are commonly available. In order to improve fault location when only limited recorded data are available, the "waveform matching" based method may be used [5]. In order to utilize this method the most, genetic algorithm based approach is used. As a result, it solves the problem of accurate FL when available data is recorded sparsely.

A. Description of the Algorithm

If power grid status, as well as FL and fault resistance are known to the short circuit program, simulated waveform will completely match with recorded waveform for corresponding fault case. Waveform matching approach is based on idea to compare recordings of faulted event against simulated recordings across same power grid. By posing fault at different locations, different simulations are obtained. The one that matches the recorded fault event the best reveals FL. Value of (4) represents the matching degree of the comparison [5].

$$f_c(x, R_f) = \sum_{k=1}^{Nv} r_{kv} \left| V_{ks} - V_{kr} \right| + \sum_{k=1}^{Ni} r_{ki} \left| I_{ks} - I_{kr} \right| \quad (4)$$

Where,

$f_c(x, R_f)$ -the cost function using phasors for matching

x, R_f -the fault location and fault resistance

r_{kv}, r_{ki} -weights for the errors of the voltages and currents respectively

V_{ks}, V_{kr} -simulated and recorded during-fault voltages respectively

I_{ks}, I_{kr} -simulated and recorded during-fault currents respectively

N_s, N_r -the numbers of selected voltage and current phasors respectively

k -the index of voltage or current phasors

B. Implementation and use of the genetic algorithm

Accuracy of waveform matching method can be drastically influenced by accuracy of performed simulation, as well as the algorithm used for posing faults for next iteration of matching.

In new approach presented in section III power flow and short circuit study are performed by using PSS/E Short Circuit program as Fig. 2 shows. In order to obtain system model that reflects status of the power grid new solution proposes use of EMS SCADA PI Historian. This tool is used for obtaining the latest load, branch and generator data in order to update power system model before simulation is performed.

From (4) we notice that cost function will be zero if phasors obtained from simulated waveforms completely match phasors obtained from recordings. It is obvious that the best fault location is found as a global minimum of (4). Therefore FL estimation problem can be translated into optimization problem. An optimal way for posing faults based on GA is used. Block diagram of this method is shown in Fig 6. In order to utilize GA, minimization problem is converted into maximization problem as shown in (5).

$$f(x, R_f) = C_{\max} - f_c(x, R_f) \quad (5)$$

Where,

$f(x, R_f)$ -the fitness function,

Cmax –maximal fitness value in the current population

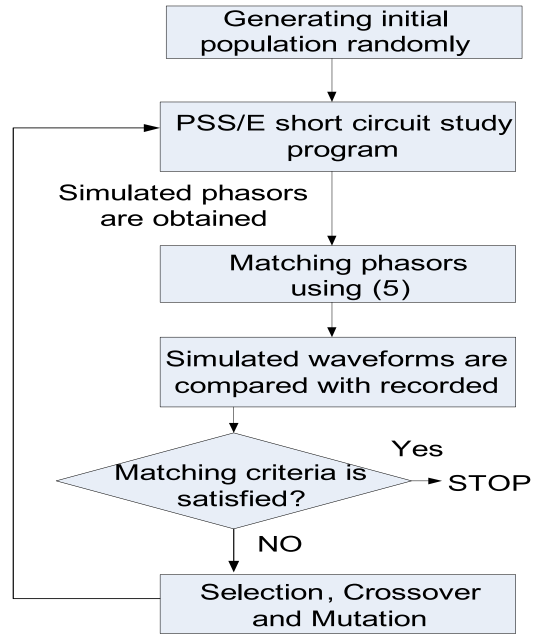


Fig. 6. Waveform matching block diagram

From (5) we see that x and R_f are selected as two variables, represented as binary strings in GA. By using three GA operators optimal fault posing for next iteration of matching is implemented [5]:

a) Selection operator mimics the process of natural selection where the fittest members reproduce most often.

b) Crossover operator, applied with probability, acts on a pair of selected members providing the exchange of binary strings.

c) Mutation operator, applied with probability, affects the single bit in a member.

VI. Phasor – based single ended algorithm

In the most common situations recorded data are available only from one end of a line, so one-end FL algorithms are used in that case. One of the well-known algorithms of this type is presented in [6]. Since this algorithm had several constraints like necessity of having prefault current recordings or assumption of constant fault impedance, which is not always true, it was necessary to develop better one-end FL algorithm. One-end FL algorithm using symmetrical components removed some obstacles [1] of this algorithm. In general, these algorithms require relatively simple calculation and their implementation is not tedious. Their accuracy depends on the simplified assumptions, but still in some cases one-end algorithm could be the optimal FL algorithm depending on available data.

VII. Optimized Fault Location Algorithm

Beside measured fault recordings most FL algorithms need some additional data. Some of them require knowledge of fault type, other require faulted line information. The idea of proposed OPFL is to find out the most suitable FL algorithm to be used in order to estimate FL from available data.

A. Description of the Algorithm

Architecture of proposed solution discussed in section III already mentioned external tools proposed by new the approach. These tools are responsible for providing initial data for FL algorithms. There are three kinds of input data available from external tools that could influence FL:

a) New event recordings consisting of either synchronized or unsynchronized samples from one or more locations.
b) Power grid information (transmission line parameters, topology information etc.)
c) Preprocessed information (fault type, fault resistance, initial estimation of fault location)

By processing input data optimal FL algorithm is chosen. In case when data from two ends of faulted transmission line are available two-end FL algorithm as the most accurate and should have priority. Otherwise it is checked whether data from one end of faulted line is available. In case of two-end algorithm, if input samples are synchronized, synchronized sampling two-ended algorithm is the most appropriate. Otherwise unsynchronized sampling two-ended algorithm is the most suitable. Similar logic is applied further and as conclusion block diagram shown on Fig. 7 is developed.

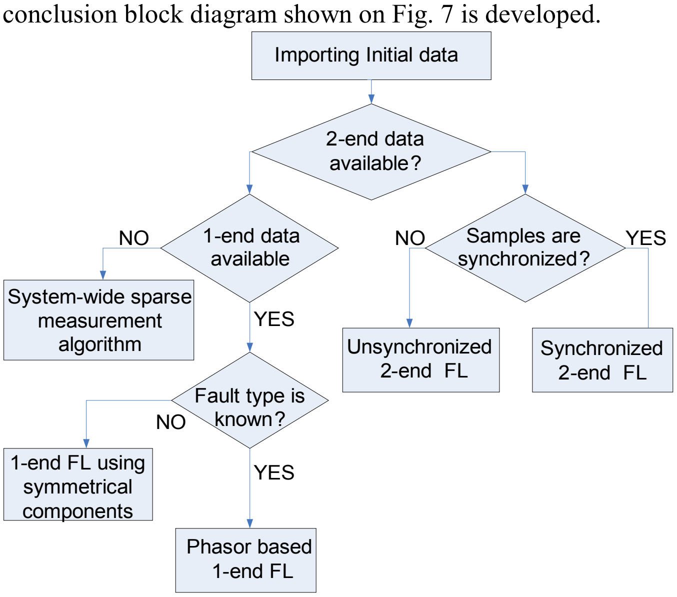


Fig. 7. OPFL algorithm block diagram

It should be noticed that in some cases multiple algorithms are applicable and it would be interesting to check how averaging results from different FL estimations using different weight functions for different algorithms could influence the results. In order to enable easy way for further testing and enhancing OPFL algorithm, decision tree is used for implementation of this algorithm, which is discussed next section.

B. Implementation of OPFL algorithm

Algorithm shown in Fig. 7 can be divided into two modules a) importing intial data and b) forming binary decision tree. It can be seen that first module provides operands and second module defines how operands are manipulated. In order to make complete algorithm transposable and readable both modules are implementined using XML. XML provides a text-based means to describe and apply a tree-based structure to the information. Implementation of each module is presented further.

For obtaining initial data used by FL algorithms, different parts of program are used. For example information about parameters of available recordings is obtained after processing input waveforms, while fault type is read directly from DFR Assitant's fault report. In order to make initial data transparent to possible changes of how they are retrived and calculated, unique XML object of initial data is implemented.

Next step was to implement easyly readable and transposable decision tree. Each node of this tree is represented with operand 1, operand 2, and operation between them, pointer to the next node if operation is satisfied and pointer to another node if operation is not satisfied. It is important to notice that names used as input operands for any node correspond to the name of some attributes in initial data object.

Once initial data object and decision tree are created, both initial data object and decision tree will be imported into program simply by calling processing engine. Engine will process decision tree in a binary format node by node while it comes to the last leaf of the tree. Each node is processed by retrieving value of operand 1 and operand 2 from initial data object and then comparing them according to the specified operation. Depending whether corresponding operation is satisfied or not the next node is chosen. Fig. 8 demonstrates this approach.

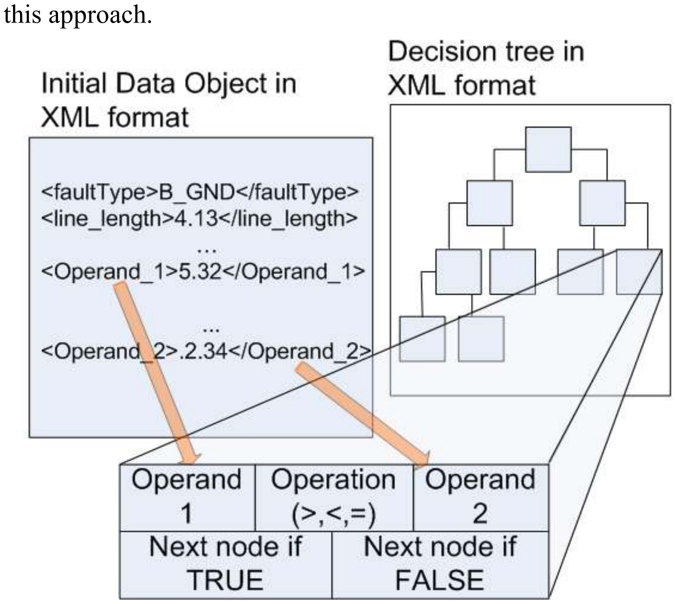


Fig. 8. Decision tree implementation

It can be noticed that this solution brings huge improvement in the process of defining and testing rules. Both initial data and decision tree structures can be easily changed and without any knowledge of programming understood. In the future this solution can easily be expend to graphic form, so that user can visually set values of decision tree. The FL software prototype was developed and set-up for specific electric power system data. This system has thirty-three substations equipped with digital fault recorders (DFRs). An automated system capable of processing, analyzing and archiving DFR data is installed and is being upgraded. Although it was not possible to automatically retrieve data, description of 15 real cases was manually furnished by utility. Automated FL analysis was run for the cases where the actual fault location was known ahead of the time. In all cases only FL algorithm using sparse measurements was applicable. Other algorithms were tested using simulated cases. Run time of complete FL analysis consisting of processing fault event recordings and other input files, correlating recordings that belong to same event, and executing applicable FL algorithm lasts 5 to 10 seconds. Only in the case of sparse measurement algorithm analysis lasts up to several minutes. Processing time of this algorithm depends on the number of input files. In the case when two recordings are available it takes about 3 minutes for calculation. This processing time is mostly influenced by the need to access external application, namely PSS/E Short circuit program, several times during the processing. More data from real cases are expected to be obtained and used for further testing, which will be reported in the future papers in more details.

VIII. Conclusion

This paper presents new architecture that makes utilization of different FL algorithms possible. This approach has modular structure, which can be extended with new techniques as they come.

Several FL algorithms and use of intelligent techniques are presented. First, synchronized sampling two-end algorithm is described and NN technique implementation used to enhance this algorithm is shown. Then use of genetic based algorithm for system-wide sparse measurement algorithm is explained. This is one of rare algorithms capable of estimating FL out of sparse measurements. It is based on waveform matching principle and by updating power grid status through EMS SCADA PI Historian as proposed. With the new approach this algorithm is drastically enhanced.

Different FL algorithms are combined in order to achieve the best FL estimation. Decision engine is implemented by using decision tree in XML format. This solution could be developed further into a graphic form, so that user can visually add new nodes to the tree and set corresponding operations and operands without any knowledge of programming.

Besides applying the most suitable algorithm on available recordings that correspond to the same fault event, this approach speeds up centralized FL procedure. Complete process from retrieving and preprocessing row data to calculating FL is done automatically which drastically decreases computation time. On the other hand the automation has made the program robust and immune to human errors.

IX. Acknowledgment

The authors wish to thank several colleagues that worked on some of the developments reported in this paper. Their names are given in the references cited with this paper. Special thanks are due to Mr. D.R. Sevcik form CenterPoint Energy for providing some field recordings and model information for the simulations presented in this paper.

X. References

[1] M. Kezunovic, B. Perunicic, "Fault Location", *Wiley Encyclopedia of Electrical and Electronics Terminology*, Vol. 7, pp. 276-285, John Wiley, 1999.

[2] M. Kezunovic, "Future Uses of Substation Data," *7th International Conference on Advances in Power System Control, Operation and Management - APSCOM 2006*, Hong Kong, October/November 2006.

[3] M. Kezunovic, B. Perunicic, J. Mrkic, "An Accurate Fault Location Algorithm Using Synchronized Sampling," *Electric Power Systems Research Journal,* Vol. 29, No. 3, pp. 161-169, May 1994.

[4] Novosel D., Hart D.G., Udren E., Garitty J.," Unsynchronized two-terminal fault location estimation ", Power Delivery, IEEE Transactions on Volume 11, Issue 1, Jan. 1996, Page(s): 130 - 138

[5] M. Kezunovic, Y. Liao, " Fault Location Estimation Based on Matching the Simulated and Recorded Waveforms Using Genetic Algorithms," Development in Power System Protection, Amsterdam, The Netherlands, April 2001.

[6] T. Takagi, Y. Yamakoshi, M. Yamaura, R. Kondow, T. Matsushima, " Development of a New Type Fault Locator Using the One-Terminal Voltage and Current Data", IEEE Trans. on Power App. and Sys., Vol. PAS- 101, No. 8, Aug. 1982 pp 2892-2898.

[7] Test Laboratories International, Inc.: "DFR Assistant - Software for Automated Analysis and Archival of DFR records with Integrated Fault Location Calculation", [Online]. Available: *http://www.tliinc.com*

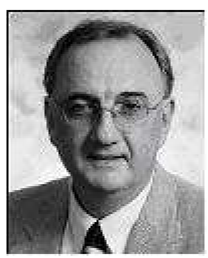
[8] IEEE Inc., 1999 "IEEE Standard Common Format for Transient Data Exchange(COMTRADE) for Power Systems", IEEE Std. C37.111-1999

[9] PSS™E Load Flow ©Siemens Power Transmission & Distribution, Inc., PTI, Schenectady, NY, USA

[10] M. Kezunovic, S. Vasilic, F. Gul-Bagriyanik, "Advanced Approaches for Detecting and Diagnosing Transients and Faults," *Med Power 2002*, Athens, Greece, November 2002.

[11] S. Vasilic, M. Kezunovic, "An Improved Neural Network Algorithm for Classifying the Transmission Line Faults," *IEEE PES Winter Meeting*, New York, New York, Jan. 2002.

XI. Biographies



Mladen Kezunovic (S'77-M'80–SM'85–F'99) received the Dipl. Ing., M.S. and Ph.D. degrees in electrical engineering in 1974, 1977 and 1980, respectively. Currently, he is the Eugene E.Webb Professor and Site Director of Power Engineering Research Center (PSerc), an NSF I/UCRC.at Texas A&M University He worked for Westinghouse Electric Corp., Pittsburgh, PA, 1979-1980 and the Energoinvest Company, in Europe 1980-1986. He was also a Visiting Associate Professor at Washington State University, Pullman, 1986-1987. His main research interests are digital simulators and simulation methods for relay testing as well as application of intelligent methods to power system monitoring, control, and protection. Dr. Kezunovic is a member of CIGRE and Registered Professional Engineer in Texas.

Maja Knezev (S'03) received her Dipl. Ing. degree in electrical engineering from the University of Novi Sad, Serbia, in 2004. Since 2005 she has been a M.S. student in the Department of Electrical Engineering, Texas A&M University, College Station. Her research interests include fault location, power system monitoring, protective relaying and substation automation.