

# Localization of Troubles in Telephone Cable Networks

Chuxin Chen, Teresa L. Hollidge<sup>1</sup>, and DD Sharma  
Pacific Bell

2600 Camino Ramon, San Ramon, California 94583  
cxchen@srv.PacBell.COM, thollidge@tibco.com, dxsharm@srv.PacBell.COM

## Abstract

This paper describes an intelligent system which detects the location of troubles in a local telephone cable network. Such a task is very challenging, tedious, and requires human experts with years of experience and high analytical skills. Our system captures the expertise and knowledge required for this task, along with automated access to database systems, so that the system can help a human analyst to pin-point network trouble location more efficiently and accurately, ultimately reducing the cost of maintenance and repair. The system utilizes probabilistic reasoning techniques and logical operators to determine which plant component has the highest failure probability. This is achieved by building a topology of the local cable network, constructing a causal net which contains belief of failure for each plant component, given their current status, history data, cable pair distribution, and connectivity to other components. The Trouble Localization (TL) Module described in this paper is a crucial part of a larger system: Outside Plant Analysis System (OPAS) which has been deployed Statewide for over nine months at Pacific Bell PMAC centers. The TL system module utilizes AI and Object-Oriented technology. It is implemented in C++ on Unix workstations, and its graphical user interface is in an X Window environment.

## Background

Pacific Bell has the largest and most complex telephone cable network in the state of California. The maintenance of this network is divided into two primary areas: reactive and proactive/preventative maintenance. Reactive maintenance is customer driven. For example, when a customer is experiencing trouble, they call for assistance and a technician is immediately dispatched to resolve the problem in the quickest way possible, which is not necessarily the most cost effective. Preventative maintenance is the ongoing repair and upkeep of the network which occurs prior to affecting customers. The trigger for preventative maintenance is based on several factors, but primarily recommendations from the technicians in the field and huge quantities of data regarding the maintenance and trouble history of each segment of the entire network. The ultimate goal of maintenance is to provide the customer with clear, continuous service and preventative maintenance is critical to achieve success.

The task of preventative/proactive maintenance for the entire network is distributed over five PMACs (Preventative Maintenance Analysis Centers) at various locations throughout the State. Obviously the cost of

maintenance is very high and it is more effective to repair a troublesome network component *once* than to dispatch several technicians over time to patch and repatch the same trouble spot. Unfortunately, it isn't always easy to identify the exact location and cause of a particular source of trouble. It is a very time intensive, laborious manual task. If we are able to detect a problem early and identify the faulty device precisely, we are able to cost effectively manage the repair of the network facility before the customer is impacted.

The crucial task of a PMAC analyst is finding the exact location and root cause of telephone cable network troubles. Such a task is very challenging, tedious, and requires human experts with years of experience and analytical skills. An automated system with efficiency and precision will significantly reduce the maintenance cost and enhance customer service.

## Problem Domain Description

The goal of preventive maintenance is to identify potential troubles before they affect a customer's telephone service. Indications of a degradation in the network are buried in several sources of information or databases. It is quite tedious and time consuming to retrieve this information, correlate potential symptoms, and derive a diagnosis. Even if this massive data mining task can be performed, one has to balance the cost of exhaustive diagnosis with the timeliness of identifying more serious troubles. In practice an analyst's attention is drawn to trouble based on information in a report called Program Scan. Every night an automated testing machine (called Predictor) tests the telephone network and collects the results in the Program Scan morning report.

Preventative maintenance often begins with a series of trouble indicators from Program Scan reports, in which trouble codes are associated with telephone cables and pairs. The type of trouble code represents the integrity of the line, ranging from Test OK, DC Fault, Open Circuit, Cross Talk, Ground, or Short.

Figure 1 depicts a sample telephone network, from an exchange (CO, the central office) to a customer's neighborhood, depicted by the letters A through H. The network from CO to a customer's house consists of many network elements such as the feeder cables (06, 36), cable splices in man holes (MH#2), a cross connect box (also called a B box, labeled X in Figure 1), distribution cable (0601), several splice points on the distribution cable, and terminal boxes A through H. When a Program Scan shows a trouble, in principle the trouble can be anywhere from

<sup>1</sup> Teresa L. Hollidge is now with TIBCO Inc., (formerly Teknekron Software Systems Inc.). Her e-mail address is thollidge@tibco.com.

the CO to the wires connecting terminal boxes to a customer's house. Much of this network is buried and not easily accessible for testing. The difficulty is to locate one cable pair (one line going to a customer's house) or a small group of cable pairs from a potentially large set of cable pairs (usually 3600) requiring attention and furthermore, to find the location of the trouble along that pair or pairs. Using Program Scan data it is easy to narrow down the defective cable pair range. However, to provide redundancy or to support growth, a cable pair may appear in multiple locations (though the telephone connection is only active in a single location). In this situation it is common for a trouble at one location to manifest itself at another location. When a single trouble manifests itself in multiple locations, it further complicates the diagnosis. In summary, the problem is to find the physical location of the trouble and the associated cable pair. For example, in Figure 1, the physical location of a trouble can be any one or more of the following: serving terminals A through H, the distribution cable (bounded by cable pair count), the splice junctions, the B box, feeder cable, or the central office (CO).

The analyst looks for patterns in the data and attempts to group the problems into network segments. He then retrieves related records from various legacy systems, such as an ACR report from LFACS (a type of cable record report from a legacy system which stores information about the physical outside plant), Repair History from MTAS (another legacy system which stores trouble repair history), and a Defective Pair report from LFACS. He may also search for related P3028 reports (which are written by technicians about observed conditions in the field), study the corresponding cable map, and even run other tests to gather further information. The goal is to pin-point the trouble location so that a technician can be dispatched for field testing. This process is slow, error-prone and tedious, requires special experience, knowledge, and analytical skills.

The reasoning process of an expert analyst is quite interesting. Normally, the analyst does not have all the relevant information available up front to support a proper diagnosis. As the analysis progresses, the analyst is driven towards various plausible causes and decides to collect information from various sources. The quality of information is often poor, forcing the analyst to seek other sources of information or derive partial conclusions. Formulating plausible hypothesis and collecting data to substantiate the hypothesis goes hand-in-hand - often the new data can rule out an earlier hypothesis or point towards a more appropriate hypothesis that subsumes many possible causes. For example, if many sets of cable pair coming from a B box show trouble then it is a plausible conclusion that the trouble is in the B box and not in the individual cable pair sets. It is obvious that the reasoning is inherently non-monotonic and the traditional backward chaining techniques (as used in MYCIN) are not appropriate. What is needed is a truth maintenance system reasoning engine with the power to handle probabilistic reasoning. Causal networks combined with Bayesian inference provides an appropriate framework for non-monotonic reasoning coupled with probabilistic reasoning. Given the heavy emphasis on conditional reasoning by expert analysts, poor quality of information in databases,

and the non-monotonic nature of the reasoning, we found the Causal Network approach to be both elegant and scalable and selected it as a foundation for solving the trouble localization problem.

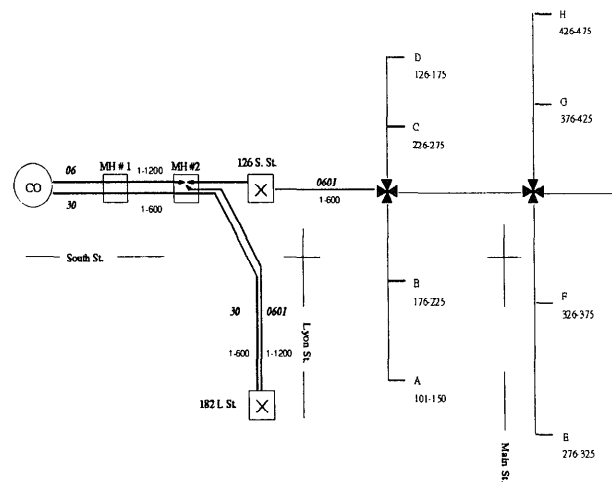


Fig. 1. An example of local telephone network.

## System Architecture

The purpose of the Trouble Localization Module is to automate the physical and mental process of a PMAC expert, capture the expert's knowledge, and provide a useful tool for both new and experienced analysts. TL provides a way for the PMAC analyst to locate cable network troubles more efficiently and accurately.

The TL module utilizes Object-Oriented technology for knowledge representation and Artificial Intelligence methodology for reasoning and inference.

The major components of the TL module and its connections with OPAS Data Server are depicted in Figure 2. The functionality of the Data Server is to access the OPAS database, interface with Pacific Bell's legacy systems, and provide the TL module with its data retrieval and storage requirement.

The TL User Interface is an X Window based GUI, which allows the user to select a trouble list from legacy system reports, determine history range, run the trouble diagnosis, display the topology, causal net, and localization results, in addition to performing disk file manipulation tasks.

The TL system module consists of three major components:

1. Trouble Localization Specialist, which is responsible for generating the topology and the causal net, quantization of Program Scan and history data, and report creation.

## Trouble Localization Strategy

The theoretical foundation of causal networks and its associated Bayesian operators is based on the book entitled *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, by Judea Pearl (Pearl 1988). The trouble localization task starts with the generation of a local plant topology graph which is very much like the hard copy cable map the PMAC analyst uses. Then another graph, a causal network, is constructed based on the plant topology. This graph takes consideration of the causal relationships among the different types of plant components, their history data, and Program Scan data.

The probability of failure is first calculated for each node, and more importantly, the evidence (belief of failure or non-failure) is propagated throughout the causal network according to certain rules. The value of belief in each node is then updated. When the causal network reaches its equilibrium state, i.e., no more evidence propagation needs to be performed, then the node with the highest probability of failure is the most likely the source of trouble.

The process starts with the creation of a trouble list which consists primarily of the Program Scan data. The cable and pair range of the data is determined by the user. Based on this input data, the TL module then requests several LFACS reports, parses them, and constructs a cable network topology. This network topology contains information on connectivity of the portion of the network of interest, as well as detailed information, including history, about each individual plant component.

In the next step, the TL module applies six operators to the given data. The first is the probabilistic reasoning (Bayesian) operator, and the remaining five are logical operators. A Causal Network is built based on the topology for applying Bayesian operator. It is a network of nodes, where each node contains state information (belief of failure) of a plant node. Two nodes are connected through a causal link, which means, the state of one node is the cause of the state of the other.

The belief of failure computed from the Causal Network is purely dependent on the Program Scan and history data, which is quantized to be used in the Bayesian inference procedure. Therefore, quantization of this data is also an important step of the process.

There are other factors which will influence the decision of which plant component is the "guilty one". Thus, in addition to the Bayesian Operator which is based on a probabilistic model, five logical operators are created to fine tune the trouble localization process. These are:

(1) TDE (Total Damage Explained) operator, which calculates the percentage of total trouble coverage in a particular plant component;

(2) DWTC (Damage Within Total Count) operator, which calculates the percentage of trouble indicators within a total cable pair count of a particular plant component;

(3) Worker operator, which calculates the percentage of working lines within the total number of cable pairs in a serving terminal,

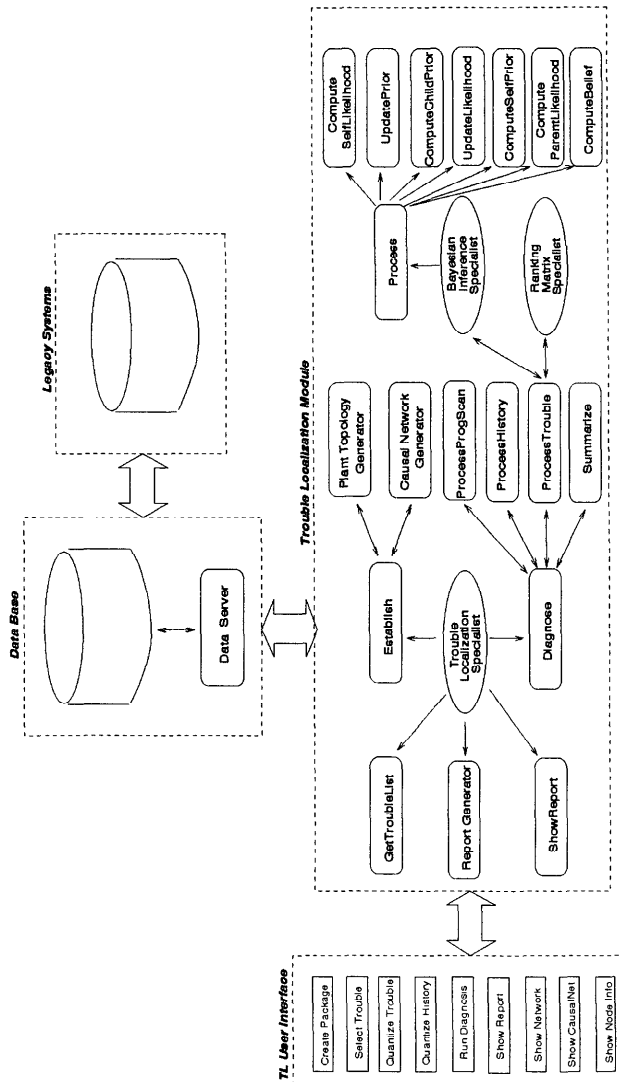


Fig. 2. TL system components and its connections to OPAS Data Server.

2. Bayesian Inference Specialist, whose function is processing the prior probability and likelihood of failure for each node and propagating the evidence throughout the causal net, and computing the final belief of failure for each plant component.
3. Ranking Matrix Specialist, which is responsible for activating logical operators and combining their results with the result from the Bayesian operator, and then generating the final trouble location ranking list.

(4) Failure Type operator, which calculates a weight factor for a particular plant component, using the weight table for failure type and the type of trouble indicators observed, and

(5) Cluster operator, which calculates the number of clusters, cluster compactness, and overall cluster measure for F1 and F2 cables.

The Failure Measurement Matrix is generated from the output of these six operators. These outputs, normalized to values between 0 and 1, form the columns of the matrix, and the plant components, identified in the network topology, form the rows of the matrix. An ordering procedure will be designed to generate a final ranking list for plant components based on the total failure belief. Several schemes are being considered:

(1) Fixed ordering, in which the order of applying the operator's output in the ranking process is fixed before run-time,

(2) Dynamic ordering, in which the order is determined during the run-time with predefined rules, and

(3) Rule-based ordering, in which the order is determined using AI rule-based techniques.

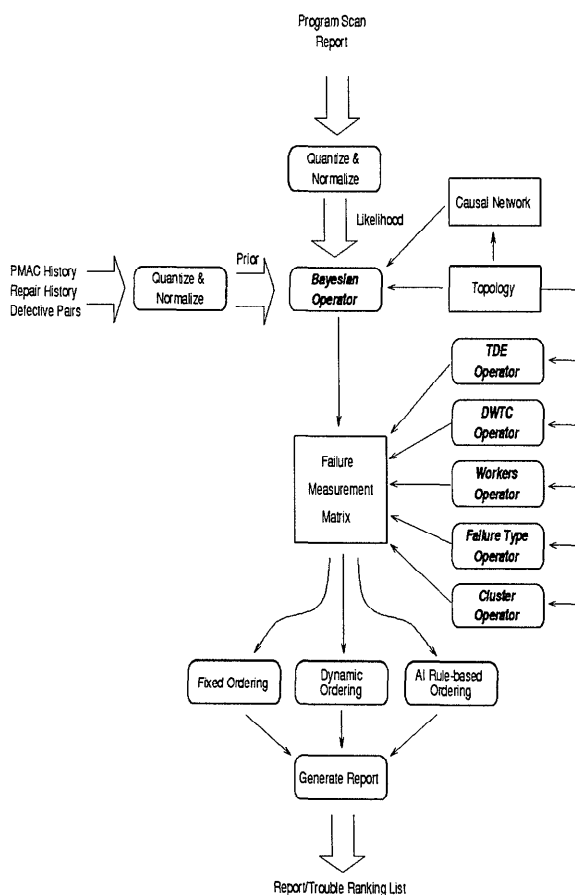


Fig. 3. The flow of control in TL module.

The following subsections further describe the processes within the TL module.

#### A. Construction of Network Topology

The goal of this procedure is to generate a network topology by issuing a request to the LFACS legacy system, receiving a report, parsing the report, and then building the topology. It is assumed that the PMAC analyst running the Trouble Localization module determines which wire center, cable, and pair range they want to examine. Using these as input data, several reports can be obtained to construct a network topology. Figure 4 shows an example, which is based on Figure 1 shown in the first section.

In the future we hope to streamline this process through direct access to the LFACS system. Currently we are constrained to parsing reports due to the proprietary nature of the LFACS database.

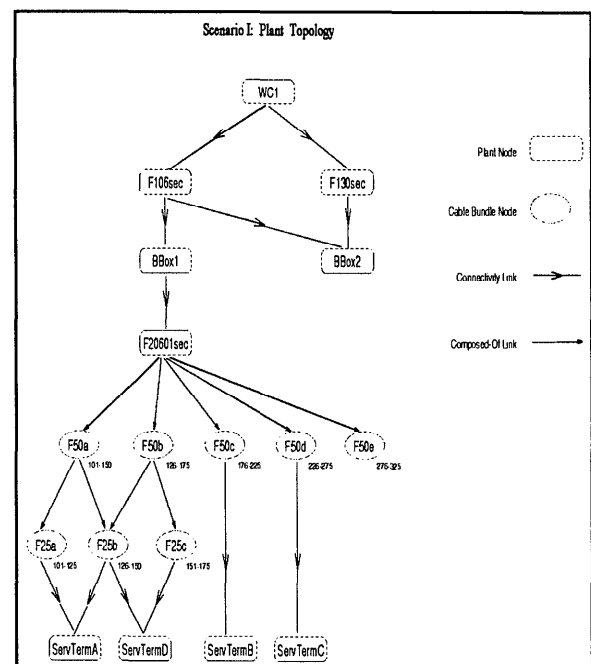


Fig. 4. An example of plant topology.

#### B. Construction of Causal Network

The Causal Network is built based on plant topology information. The graph nodes are classified into four categories:

- History Bayesian Nodes
- Plant Bayesian Nodes
- Cable Bundle Bayesian Nodes
- Program Scan Bayesian Nodes

The information in History nodes comes from three sources (from three databases):

- PMAC history,
- Repair history, and
- Defective pairs.

The information in Program Scan nodes comes from related program scan reports of corresponding cable and range. Processing is required to quantize, weight, and normalize this data so that an evidence vector may be computed for each node. The information in Plant and Cable bundle nodes is inherited from their counter parts in the plant topology.

The Causal Network (or sometimes referred as Bayesian Network) is constructed as a directed acyclic graph (DAG). The arrows in the graph represent causal influences of one node upon to another (Figure 5).

The functionality of the four different types of nodes (shown in Figure 6) in the Causal Network graph are:

**- History Node**

This node contains the probability of repeated history failure of a particular plant component.

**- Program Scan Node**

This node contains the evidence of current failure observed through Program Scan data, on a cable bundle.

**- Cable Bundle Node**

This node contains the probability of failure within a specified range of the cable bundle.

**- Plant Component Node**

This node contains the belief of failure regarding the particular plant component.

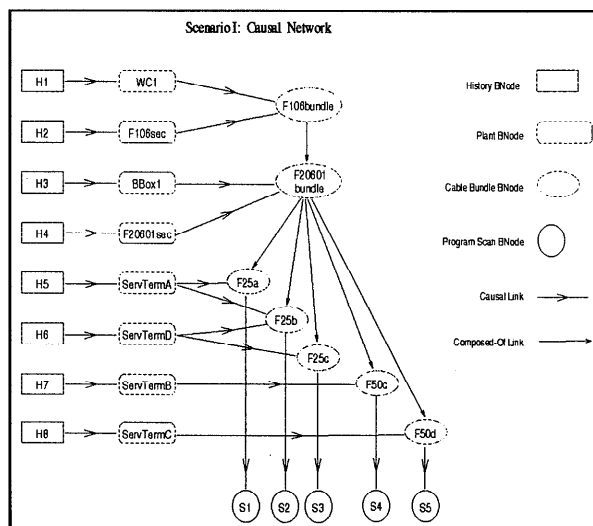


Fig. 5. The corresponding Causal Network to Figure 4.

In directed acyclic graphs, the relationship between any two nodes is defined as a parent-child relation. This is illustrated in Figure 7. The parent node is the one with a link arrow pointing away from it, and the child node is the one with a link arrow pointing towards it.

A Bayesian node  $x$  is characterized by its belief  $BEL(x)$ , the prior probability,  $p$ , and the likelihood parameter,  $l$ . New information regarding a node is communicated to it by either changing the prior probability or the likelihood parameter. After receiving the new information, new values of appropriate parameter are computed and the evidence is propagated to other neighboring nodes.

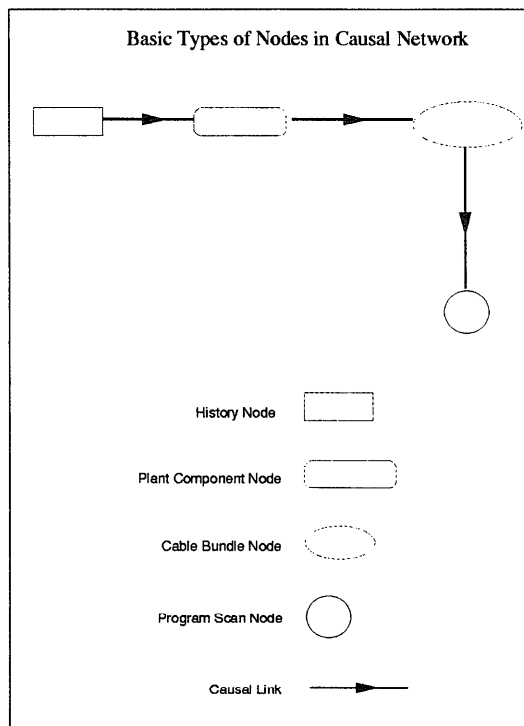


Fig. 6. Basic types of graph nodes.

The flow of propagation of evidence between a parent node and a child node is that the prior probability,  $\pi$ , is passed from the parent to the child, and likelihood,  $\lambda$ , is passed from the child to the parent.

In case of a network of nodes, the flow of propagation for each node becomes:

- Compute  $\pi$
- Compute  $\lambda$
- Compute  $BEL$  (belief)
- Propagate  $\pi$  to its child nodes

- Propagate  $\lambda$  to its parent nodes

where  $BEL(x) = \alpha \cdot \pi(x) \cdot \lambda(x)$

$\alpha$  is a scaling factor,

$$\pi(x) = \pi(u) \cdot M_{x|u},$$

$$\lambda(x) = M_{y|x} \cdot \lambda(y),$$

and

$$M_{y|x} = P(Y = y | X = x)$$

$$= \begin{bmatrix} P(y_1|x_1) & P(y_2|x_1) & \cdots & P(y_n|x_1) \\ P(y_1|x_2) & P(y_2|x_2) & \cdots & P(y_n|x_2) \\ \vdots & \vdots & \ddots & \vdots \\ P(y_1|x_m) & P(y_2|x_m) & \cdots & P(y_n|x_m) \end{bmatrix}$$

is the conditional probability matrix.

The conditional probability matrix is especially useful when there are multiple parent and child nodes.

Further details on the computation of the parameters of a Bayesian node and propagation rules can be found in Ref. [1] (Pearl 1988).

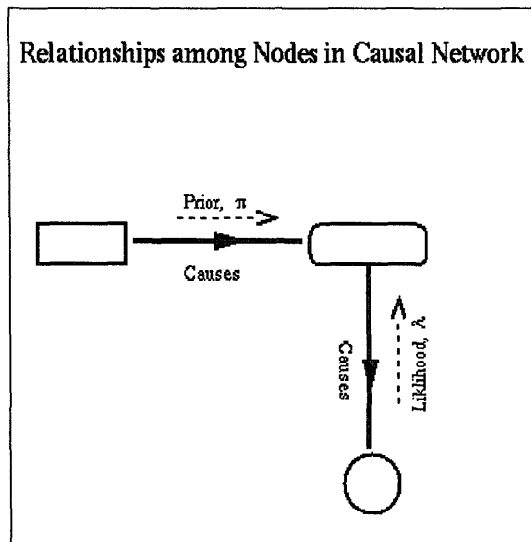


Fig. 7. Inter-node relationships.

### C. Quantization of Program Scan Data

The goal of this procedure is to compute the values of likelihood in the Program Scan nodes of the Causal Network (Figure 8). The input data is obtained by running Program Scan, given the cable and its pair range, which

provides the ver code, trouble message, and telephone number for each cable pair. The ver code is then mapped to a number between 0 and 1 based on a table look-up. The Program Scan data is divided into groups according to the cable pair range of Cable Bundle node, to which a Program Scan node is connected. Within a group, all values of mapped ver codes are summarized and normalized to calculate the likelihood vector for a particular Program Scan node.

### D. Quantization of History Data

The history information related to a particular plant component is classified into three categories:

- PMAC History
- Repair History
- Defective Pair History

The information in each category is retrieved and processed, and a number between 0 and 10 is assigned. The number then is multiplied by a predetermined weight factor for that category. The resulting values from the three categories are summarized and normalized to a single value which is used to compute the prior (evidence) vector (Figure 9). This vector is stored in the corresponding History node in the Causal Network.

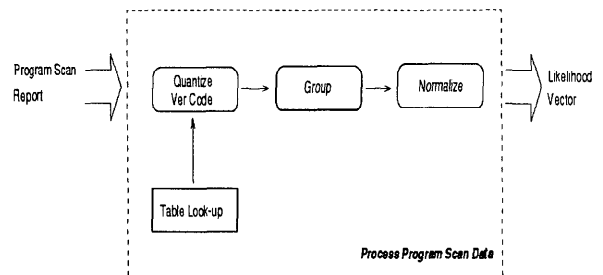


Fig. 8 Program Scan data quantization.

#### D.1 Quantize PMAC History

The source of PMAC History will be mainly from P3028 reports. These reports are classified into six different bins:

- Input file
- Closed; Construction job
- Pending status; No work needed but trouble found
- Closed; No trouble found
- Work in progress
- Closed; Work completed

The value of quantized PMAC history is then selected from a table.

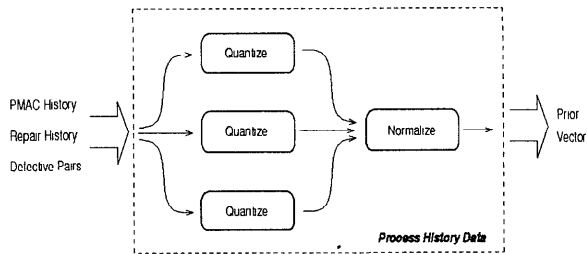


Fig. 9. History data quantization.

### D.2 Quantize Repair History

The quantization method of repair history is based on predicting the behavior of repair needs of a particular plant component from its past records. By fitting a line or curve to a set of points located within time and percentage of repair axes, the future time at which the percentage of repair will exceed a certain fixed value (e.g., 10 %), can be determined (Figure 10). Using this time value, the quantization value is then computed based on a predefined function (Figure 11).

### D.3 Quantize Defective Pair History

The quantization value for defective pairs is determined based on a simple function (Figure 12). The input data is expressed as a percentage within a cable. For example, if the following mapping function is used, then 20 % of defective pairs will result in value of 5 for the quantization.

### D.4 Combine The Quantized Histories

The overall value of history quantization is calculated by multiplying each individual value by its corresponding weight factor, adding them together and then dividing by the sum of the weight factors.

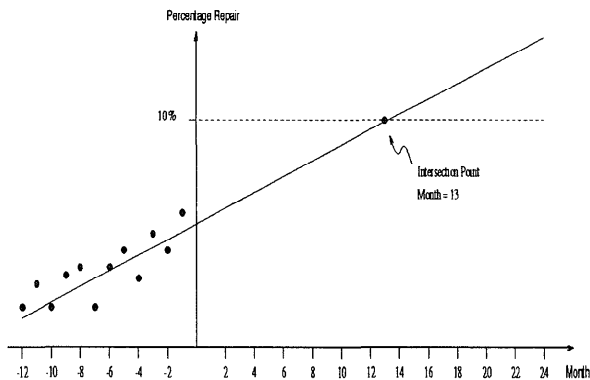


Fig. 10. Fitting a line to a set of points representing repair history.

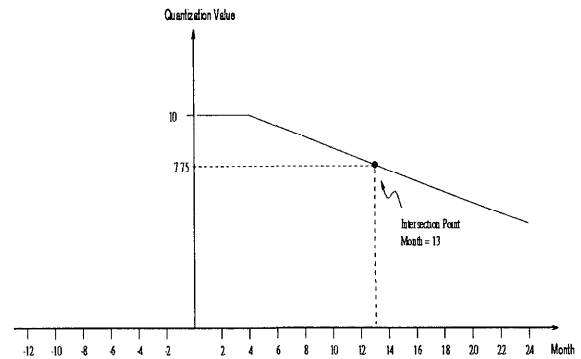


Fig. 11. Calculating the quantization value for repair history.

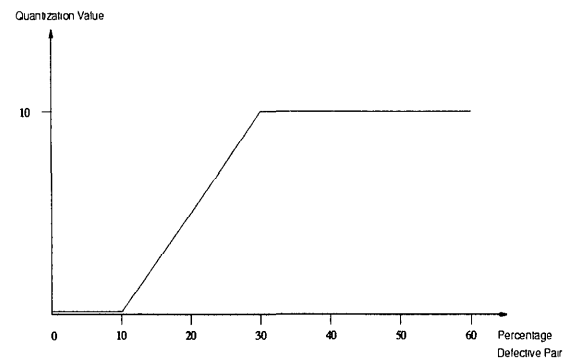


Fig. 12. Calculating the quantization value for defective pair history.

The final value of combined quantization is computed as:

$$\text{Overall Value} = (a * R + b * D + c * P) / (a + b + c)$$

where

a = Weight Factor of Repair History

b = Weight Factor of Defective Pair History

c = Weight Factor of PMAC History

R = Quantized Value of Repair History

D = Quantized Value of Defective Pair History

P = Quantized Value of PMAC History

## System Implementation

The OPAS system is deployed in two centralized data

centers for use throughout the State of California by the five PMAC centers. Each data center contains all the communications links to the legacy systems and the data storage required for the OPAS system and the TL module. In essence, the OPAS database, which has been extended to support TL, acts as part of the knowledge base for the TL Module.

The TL Module is accessed by the PMACs through their workstations and is used primarily by analysts and engineers. The system is used on as-needed basis depending on the analysis requirements of the particular PMAC. As time progresses, we would expect that the TL Module would be used on a daily basis to help identify specific trouble areas of the network.

The expected payoff from the addition of TL to the OPAS system is tremendous. The primary purpose of TL is to locate the source of network trouble. By identifying the correct source of the problem without a site visit from a technician is a considerable cost savings. Prior to the develop of this system multiple site visits were required before a trouble could be identified and then repaired. Often the location of a trouble had been guessed and a technician dispatched only to discover that the trouble was elsewhere and the technician in the field was not qualified for that type of repair. With the addition of TL to OPAS we are able to dispatch the required resources to the site with the proper equipment and get the problem resolved before the customer is affected. In addition, since more network problems can be accurately located, the analysts and engineers in the PMAC are better able to plan for the repairs (i.e. schedule digging crews, allocate funding).

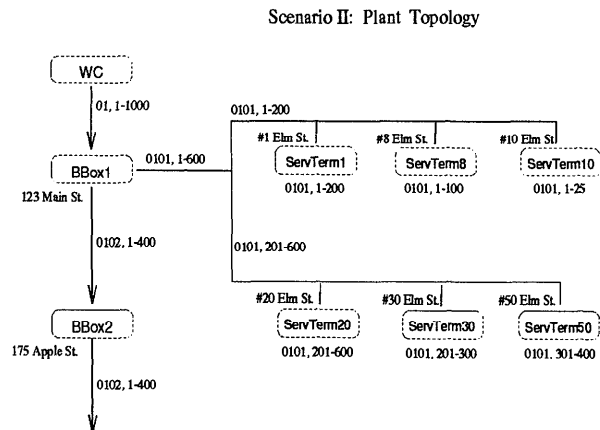


Fig. 13. Another example of local telephone cable network.

The other primary source of payoff is the reduction of the work load on the PMAC analysts. By providing the TL module, the PMAC analysts are able to bypass the horrendously tedious manual process of trouble localization analysis. Because the TL module is an automated process, the manual work is no longer necessary thereby speeding up the analysis process and freeing the PMAC analysts to

work on larger projects. In addition, TL is of great benefit in several of the PMACs where the current staff do not have the extensive outside plant knowledge and analytical skills required to do trouble localization. TL provides them with expert analysis where none is currently available. Also, because it is fully automated, TL can provide the PMAC users with fully documented support for their analysis decisions, which can be reviewed by the PMAC staff for accuracy and training.

Overtime, we expect that the use of TL within the PMAC will build the knowledge of the existing staff in analysis and localization process. With the future development of an explanation facility for each step in the analysis process, users will learn the analysis steps required and will readily apply their knowledge. And as the knowledge of the user team increases they will be able to manipulate the current weights and operators to better suit their particular environment. And of course, since we are replacing a manual process with a primarily automated process, we expect the current work will be accomplished in a timely fashion.

Name	Description	Type	In Count	Out Count	Address
WC	wire center			01, 1-1000	
BBox1	x connect		01, 1-1000	0101, 1-600	123 Main St.
BBox2	x connect		01, 1-400	0102, 1-400	175 Apple St.
F1CBL01	F1 cable			01, 1-1000	
F2CBL01 01	F2 cable			0101, 1-600	
F2CBL01 02	F2 cable			0102, 1-400	
ServTerm 1	terminal	FIXED	0101, 1-200		#1 Elm St.
ServTerm 8	terminal	RA	0101, 1-100		#8 Elm St.
ServTerm 10	terminal	FIXED	0101, 1-25		#10 Elm St.
ServTerm 20	terminal	FIXED	0101, 201-600		#20 Elm St.
ServTerm 30	terminal	FIXED	0101, 201-300		#30 Elm St.
ServTerm 50	terminal	RA	0101, 301-400		#50 Elm St.

Table 1. Plant component information.

Development of the underlying TL system was undertaken by three individuals and took approximately 36 man-months. Currently a team of five is completing the work through full-deployment which is expected within



six months. The development team relied heavily on their knowledge of AI, C++, database programming, user interface and system design. In addition, considerable effort has been spent with the domain experts to help the team understand the domain and to ensure that the system under construction would suit the needs of the users. The cost for the entire development of TL is estimated at \$1.4 million.

Cable	Pair	Ver Code	Serving Terminal	CO Side Cable
0101	5	11	servTerm10	01:1
0101	6	11	servTerm8	01:2
0101	7	14	servTerm1	01:3
0101	12	14	servTerm8	01:4
0101	20	21	servTerm1	01:5
0101	21	21	servTerm10	01:6
0101	22	21	servTerm8	01:7
0101	23	21	servTerm1	01:8
0101	24	21	servTerm1	01:9
0101	25	11	servTerm1	01:10
0101	26	14	servTerm8	01:11
0101	27	21	servTerm8	01:12
0101	28	11	servTerm1	01:13
0101	50	11	servTerm1	01:14
0101	60	11	servTerm8	01:15
0101	75	11	servTerm1	01:16
0101	79	11	servTerm1	01:17

Table 2. The trouble list from Program Scan.

## Application Results

The testing of TL module was conducted in two stages. The first one involved individual component testing, especially of the Bayesian Inference Specialist. Over sixty different scenarios were designed to verify the response and behavior of the causal net.

A point worth mentioning is that the object-oriented design approach allowed us to easily manipulate the input values and observe the output of each system component. The second testing stage involved the entire system. Figure 13 shows one of the scenarios.

Table 1 shows the plant information retrieved from the database.

**Trouble Description:** Table 2 shows that battery cross and ground trouble indicators occur on Cable 0101 between

the pair range 1 to 100, in a random fashion. However, the troubles are mapped back to CO side cable, 01, from pair range 1 to 17, consecutively.

**Desired Result from PMAC Analyst:** F1 cable 01 is the most likely cause of the trouble, because of the cluster observed.

**Actual Result from TL Module:** It is displayed in Table 3, which shows a ranked list of plant components.

**Reasoning by TL Module:** Component [f1cable01] ranks the highest because it covers 100.% of troubles, and it has the highest value in Category [Cluster] of the Failure Matrix.

The above reasoning is generated by the Ranking Matrix Specialist. This result is consistent with the expectation of PMAC experts.

Rank	Name	TDE	Belief	DWTC	Worker	Cluster	Fail Type
1	f1cable01	1.000	0.000003	0.017	0.000	0.680	0.014
2	servTerm1	1.000	0.000895	0.085	0.529	0.091	0.043
3	servTerm8	1.000	0.001960	0.170	0.353	0.086	0.086
4	f2cable0101	1.000	0.000028	0.028	0.000	0.170	0.000
5	bbox1	1.000	0.000003	0.028	0.000	0.000	0.000
6	WC-ET	1.000	0.000000	0.000	0.000	0.000	0.000
7	servTerm10	0.588	0.006157	0.400	0.200	0.080	0.232
8	bbox2	0.000	0.000100	0.000	0.000	0.000	0.000
9	servTerm20	0.000	0.000001	0.000	0.000	0.000	0.000
10	servTerm50	0.000	0.000001	0.000	0.000	0.000	0.000
11	servTerm30	0.000	0.000001	0.000	0.000	0.000	0.000

Table 3. Ranking matrix generated by the TL module.

## System Maintenance

Currently the TL Module and the OPAS system are being maintained by the developers and we are currently working on plans and procedures for moving the maintenance of the entire system to a maintenance organization within the company. The largest portion of the TL knowledge base is in the form of a large Oracle database which is within the OPAS system. The database draws data from the specified legacy systems as scheduled. Several of the legacy systems provide daily/hourly update and for others, data is requested on demand from the legacy system and

the database is updated at that time.

Since the OPAS system is applied to the current copper cable network, the knowledge about the structure of the plant components will remain virtually unchanged. Thus the knowledge related to the plant itself does not need updating once it is established. Updates are required for the decision rules that govern the determination of ranking of faulty plant devices or components. In this regard, the system is expandable because the design of Ranking Matrix Specialist allows more operators to be added to the matrix.

## Concluding Remarks

We have presented an intelligent system which utilizes object-oriented technology and a Bayesian network inference approach to solve real world problems. The system was designed to assist PMAC analysts but not to replace them; the system is to release the analysts from boring, tedious work and let people do their job in a more effective and accurate way. The system generates a ranked list of faulty components, but the final decision is made by the analyst. It is a decision support system where the user can change various parameters, try different scenarios, and compare different results to reach his conclusion.

Future development includes enabling the system with learning capability and diagnosis explanation capability. We hope to provide enhanced graphical output, so that the dispatched technician will have an annotated cable map to aid the repair and maintenance of the faulty component.

While this system was intended for the copper telephone cable network, we believe that the technology used here could be enhanced to support broadband networks.

## Acknowledgment

The authors wish to thank Sam Allen for his expert advice, Ray Ronco and Tom Gill for their support, and all the OPAS team members for their assistance. The authors also thank the anonymous reviewers for their comments.

## References

1. Judea Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publishers, Inc., 1988.
2. Grady Booch, *Object-Oriented Analysis and Design with Applications*, The Benjamin/Cummings Publishing Company, Inc., 1994.
3. Todd E. Marques, *StarKeeper Network Troubleshooter: An Expert System Product*, AT&T Technical Journal, November/December 1988, pp 137-154.
4. DD Sharma and Teresa Hollidge, *OPAS Functional User Requirements*, Pacific Bell internal documents, August 1994.
5. Chuxin Chen and Teresa Hollidge, *OPAS Trouble Localization Module, Final Report*, Pacific Bell internal documents, January 1996.