The Development of an Intelligent Alarm Processor

by

Mazen Abdallah Abdul-Rahman Salah

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

ELECTRICAL ENGINEERING

July, 1991

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

U·M·I

University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road. Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Order Number 1354075

The development of an intelligent alarm processor

Salah, Mazen Abdallah Abdul-Rahman, M.S.

King Fahd University of Petroleum and Minerals (Saudi Arabia), 1991

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

THE DEVELOPMENT OF AN INTELLIGENT ALARM PROCESSOR

BY

MAZEN ABDALLAH ABDUL-RAHMAN SALAH

A Thesis Presented to the
FACULTY OF THE COLLEGE OF GRADUATE STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE
In

ELECTRICAL ENGINEERING

JULY 1991

309

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS
DHAHRAN 31261, SAUDI ARABIA

COLLEGE OF GRADUATE STUDIES

This thesis, written by Mazen Abdullah Salah under the direction of his Thesis Advisor and approved by his Thesis Committee, has been presented to and accepted by the Dean of the College of Graduate Studies, in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE.

Thesis Committee

Al-Shehri A M
Thesis Advisor

ELAMIN, I. 27/7/91
Member

Aref, Mostafa
Member

GEORGE OPOKU
Member

Al-Shehri
Department Chairman

Dean, College of Graduate Studies

6. 10. 91
Date

This thesis is dedicated to my parents, Abdullah and Nawal.

ACKNOWLEDGEMENTS

Support of King Fahad University of Petroleum and Minerals and my employer, Saudi Consolidated Electric Company East are gratefully acknowledged.

I wish to express my appreciation to Dr. Abdullah Al-Shehri for serving as my advisor. I also wish to thank the other members of my thesis committee for their efforts on my behalf; Dr. I. El-Amin, Dr. G. Opoku, and Dr. M. Aref.

I would like to further thank Prof. Fareed Zedan for his continuous support.

Thomas Goodgame for his assistance with WordPerfect.

Ken West of SCECO SCADA Software; and Ehsan, Waseem, and Mukhtar of SCECO Power Control Center for their efforts and information.

And finally to offer thanks and respect to my wife, brothers, sister, and parents for their patience, forbearance and support.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

Page No.

CHAPTER 3 BUILDING PROCESS OF THE IAP 61

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES

LIST OF FIGURES

NOMENCLATURE

AI = Artificial Intelligence

a(v) = Accuracy of telemetry device 'v' in percentage

BRKR = Breaker

BTD = Bad Telemetry Devices

DACS = Data Acquisition and Communication Subsystems

DE = De-Energized (Bus or Transformer)

E(i) = Error value reading assumed at equipment 'i'

EMS = Energy Management System

ES = Expert Systems

Gen = Generator

HS = High side of the transformer

HV = High Voltage at a substation

IAP = Intelligent Alarm Processor

IE = Inference Engine

KB = Knowledge Base

KE = Knowledge Engineer

LHS = Left Hand Side of a rule

LS = Low side of the transformer

LOL = Loss of load

LV = Low Voltage at a substation

N(i) = Nominal or rated reading at equipment 'i'

OE = Open Ended (for the line)

OL = Off-Lone (for the generator)

OOS = Out Of Service (for the line)

PCC = Power Control Center

RHS = Right Hand Side of a rule

RTU = Remote Terminal Unit

SBO = Substation Black Out

SCADA = Supervisory Control And Data Acquisition

S/S = Substation

Trans = Transformer

خلاصة الرسالة

اسم الطالب : مازن عبدالله عبدالرحمن صلاح
عنوان الدراسة : تطوير معالج الانذارات الذكي
التخصص : هندسة كهربائية – نظم الطاقة
تاريخ الشهادة : يوليو ١٩٩١م .

موضوع الرسالة الرئيسي هو تطوير معالج ذكي للانذارات التي تصل إلى مركز التحكم خلال الاضطرابات على الشبكة الكهربائية . يعمل المعالج على عرض الانذارات (الوافدة) بطريقة أكثر إيجازاً وتعبيراً وشمولية . ويتم ذلك باستخدام الذكاء الصناعي لمعرفة مصادر الانذارات واسبابها .

في هذه الرسالة طُوِّر نظام الخبير ذو الأسس القانونية والذي يتألف من ٦١ قانون . ويستخدم هذا النظام الأداة المسماه "كليبس" والتقنيات المعروفة بـ "التسلسل الأمامي" ، ولقد أطلق اسم "معالج الانذارات الذكي (أي إي بي) على هذا النظام .

يضاهي هذا النظام مرحل متمرس في مركز التحكم بالطاقة الكهربائية في طريقة تحليلية لإنذارات نظم الطاقة للوصول إلى نتيجة عما حدث بالشبكة من إضطراب .

إن الوصول لهذه النتيجة مطلوب للاستدلال على منهج الاجراءات اللازم عملها لاعادة تقويم نظام الطاقة الى الحالة الطبيعية وبأسرع وقت ممكن .

وتتم معالجة الانذارات الوافدة في النظام المطور على أسس معرفة القوانين التجريبية المشتقة خصيصاً لتوليد رسائل الانذار المتميزة بدقة الوصف والذكاء . فضلاً عن ذلك ، يتحكم النظام بعرض المعلومات للمرحل أو كبتها عنه على مبدأ ماهو مفيد للعرض أو خلافه .

ويعتمد النظام المطور في استخلاصه للنتائج المطلوبة على بيانات التوصيل المتعلقة بالشبكة لربط علاقة المعدات ببعضها .

وبإستطاعة النظام المطور توليد الرسائل الوصفية الموجزة حتى في حالة فقدان بعض الانذارات المتوقع استقبالها في مركز التحكم نتيجة لتعطل أجهزة القياس في المحطات . هذا بالاضافة إلى إرساله جملة تحذيرية للمرحل لإعلامه بالأعطال الحاصلة على أجهزة القياس .

جامعة الملك فهد للبترول والمعادن
درجة الماجستير في العلوم

THESIS ABSTRACT

FULL NAME OF STUDENT: MAZEN ABDULLAH SALAH
TITLE OF STUDY THE DEVELOPMENT OF AN
INTELLIGENT ALARM PROCESSOR (IAP)
MAJOR FIELD POWER SYSTEM
DATE OF DEGREE JULY, 1991

The major objective of this thesis is to develop an Intelligent Alarm Processor (IAP) for processing excessive alarms at the Power Control Center (PCC) especially during a major power disturbance. The received alarms will be analyzed by the IAP and the outcome will be presented to the dispatcher in a shorter, more concise and meaningful manner.

The processing of multiple alarms into fewer concise statements is best achieved by applying artificial intelligence techniques into the received alarms. In this thesis, a rule-based Expert System consisting of 61 rules is developed. It utilizes CLIPS as a tool with a forward chaining reasoning. This system emulates an experienced dispatcher at the PCC in analyzing the power system alarms to reach a

definite conclusion. The desired conclusion describes what has happened on the network. Such a conclusion is needed to lead to the course of actions required to return the power system to the normal state as soon as possible.

The developed IAP manipulates the received alarms on the basis defined by the derived empirical rules to produce the descriptive messages. Furthermore, it controls the suppression and display of information to the dispatcher on the basis of what he needs and does not need to see. In deriving the required conclusion, the IAP relies heavily on the connectivity data of the network to relate the various equipment to each other.

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Interconnected Power System networks have grown rapidly over the last few years. Today these networks are monitored and controlled by modern Power Control Centers (PCC). The PCC operators will be alerted of any abnormality on the power system network by means of some alarm messages. Alarm processing has been a traditional feature that has not changed over several generations of the power control center design.

During some power system disturbances, tremendous number of alarms are received at the PCC. The operators are required to act very fast to return the power system network to its normal state. A lot of alarm analysis is needed to understand what has happened on the network before any action is taken. Operations personnel have often voiced the desire for a better way to monitor a power system than provided by existing alarm processing software and hardware [1]. A very promising method to improve alarm message presentation during a disturbance is to reduce the number of alarms by processing multiple alarms in a more intelligent manner. This is realized through the application of an Expert System (ES) to alarm processing, which is the subject of this thesis. Such an expert system will be called the Intelligent Alarm Processor (IAP).

In recent years expert systems technology has captured a lot of interest in many fields of electric power operation, engineering and control. This trend is likely to continue [2]. The expert systems application to alarm processing is a new field that is still under research by different electric utilities and universities. The work in this field has not yet defined a complete operational system that can be implemented in the power control centers. However, the work done so far was merely investigation and studies of the feasibility of ES application to alarms processing [3,4,5].

The power system alarms should not be simply rearranged for display. However, they should be analyzed together with the available software at the PCC to reach a conclusion on what has happened on the network in the shortest possible time. Expert systems has the ability, when programmed properly, to emulate the reasoning process of a human expert in a particular problem domain when it is encountered. Therefore, the expert system techniques were sought by many researchers as a promising solution to the alarm processing problems [1,5].

In this thesis a Intelligent Alarm Processor (IAP) prototype is developed. The IAP processes the alarms received at the PCC during power system disturbances to produce fewer intelligent and more concise messages. The messages describe what have happened on the network. The knowledge used to develop the IAP is based on the empirical rules that were derived from experienced dispatchers at the PCC. The IAP utilizes heavily the connectivity information, which is available at the PCC data base software, to relate the various power system component to each other.

This thesis is divided into five chapters. Chapter 1 is an introductory part where the applications of ES to electrical power engineering, in general, and to alarms processing in particular will be cited. This is followed by definition of the problem of alarms processing in modern power control centers. The chapter concludes by highlighting the scope of work for the thesis. Chapter 2 reviews the expert systems components, languages and tools. Special emphasis will be given on the language of choice 'CLIPS'. The second part of Chapter 2 discusses Energy Management Systems (EMS) operations and the applications of expert systems into EMS. More elaboration will be made on alarms processing in EMS and alarms types, classifications, conditions and generation. The solution techniques and building process of the Intelligent Alarm Processor (IAP) will be explained in Chapter 3. In chapter 4, the IAP operation and capabilities will be demonstrated using two different power system models. Finally, Chapter 5 will conclude the thesis with a summery and recommendations for future work.

1.2 LITERATURE REVIEW

Expert Systems design is currently a very active area in computer science. In the 1960's, this research was limited to few research laboratories in a small number of universities. Currently, there are numerous companies, universities and research laboratories who are active in expert systems research [1]. As a result, significant progress has been made in the development of the expert systems, and many expert systems have been successfully built for various applications.

One of the widely known applications of Expert Systems (ES) is "MYCIN" which was developed for medical diagnosis and therapy by Stanford University in 1977 [6,7]. The MYCIN system interviews physicians and receives, an input, answers to questions posed by the system, then outputs ordered set of diagnosis and therapy recommendations. The architecture of the system is rule-based, exhaustive backward chaining with uncertainty and uses LISP as a tool.

In 1981-1982 another successful application of expert systems was developed by Carnegie-Mellon and Digital Equipment Corporation. A rule-based expert system "R1" was designed for configuring VAX computer subsystem. It utilizes OPS5 as a tool and forward chaining techniques [7,8]. R1 configures the VAX computer systems by projecting the need for subassemblies given a high level description of the system.

For the analysis and synthesis of electric circuits, "EL" and "SYN" expert systems were proposed. More information about these expert systems can be found in references [7] and [9]. Further, reference [10] presents an expert system scenario for computer-aided control system design.

The application of Expert Systems to power systems is a new area of research. From the power system point of view, many areas of electric power systems analysis, planning, operation, and training appear to be promising fields for ES application [7,11]. This is due to the fact that the behavior of modern interconnected power systems has become more and more complicated, causing the decision-making process to be increasingly difficult. In addition, it takes a relatively long time to gain the knowledge needed for system planners, operators and analysis.

Expert Systems can be developed to incorporate the existing knowledge in power systems planning and operation into more advanced and intelligent programs. They can also offer assistance to system operators during a time-critical situation and provide a valuable training tool [7]. The most promising fields of application in the area of power systems were identified as follows [4]:

o Contingency analysis - a more robust selection of cases to be studied, an evaluation of the solution accuracy of using a steady state algorithm instead of a dynamic stability algorithm, and a more concise presentation of the solution results.

- Alarms processing - a more concise statement of the problem and to provide a priority to the importance of each alarm. An elaborate discussion of this field of application will be presented in the next Chapter.

- State estimation - a more complete method of bad data and of biased data identification, an alternative method of tap estimation, and to adapt the bus section load models of conforming and of non- conforming loads.

- External model estimation - the buses which should be reduced, the pseudo measurements which are the best indication of the state of other power systems, and an analysis of the present state of the total power system.

- Remedial action - the controls which should be considered, the control order (generation or tap position changes) and the controls which should be used after the next contingency.

- Automatic generation control-selection of regulation participation factors for units under control, unit not responding logic, and automatic tuning of unit and system parameters.

- Economic dispatch-generation of energy conversion curves, selection of curves, more regions and valve points.

- Unit commitment decision - when to re-execute the algorithm, selection of algorithm (e.g., full, sequential or truncated dynamic programming) selection of constraints to be enforced, and models to be used (e.g., linear, quadratic, or piece-wise linear energy conversion curves).

- Short-term load forecast - selection of base curve, selection of weather variables, selection of model parameters (e.g., degree of differencing, number of components in a multiple autoregressive moving average process), and estimation of the impact of exogenous variables (e.g., Nielsen ratings).

- Interchange evaluation - selection of possible schedules to be evaluated, pricing of potential schedules, and execution of an evaluation algorithm (e.g., economy A or B).

- Dispatcher's optimal power flow - selection of control variables to use, selection of control variable ordering, and selection of constraints to enforce.

- System restoration - selection of cranking path(s), selection of load restoration, selection of generation restarting schedule.

- Contingency relay arming - selecting of relays to be armed or disarmed, selection of key system parameters (equipment status) to be used as triggers, selection of equipment(s) to be removed for each contingency.

- Energy cost reconstruction - selection of costing algorithm, selection of unit(s) to be used for costing, selection of transaction schedules to be used for costing, and selection of transmission loss model for costing.

- Load shedding - selection of distribution feeders to open which will most likely remedy system overload or voltage violation.

- Load management - selection of load curtailment strategy based upon expected load and weather trends.

- Trouble call analysis - faster identification of outaged equipment based upon equipment reliability, weather conditions and load demand in addition to customer complaints.

A summery of the recent advances in this field is cited below.

In the area of power systems operations planning, R. Fujiwara, et al, [3] in 1986, described the details of an intelligent load flow engine. The objective of the engine was to provide both an expert and elementary user with a friendly working environment, aiming at enhancing a user's creativity in making operational plans. The knowledge for the expert system was acquired from the experienced power system planners and implemented in PROLOG. Besides the expert system, the intelligent load flow engine realized a graphic user interface, which was composed of a power system map and control panel, and one-touch data retrieved for fast interaction. A sample system having 54 generators, 261 buses and 289 lines was tested by the intelligent load flow engine. It showed satisfactory results with some limitations. The limitations concerned editing rules, rules explanation, communication, rules consistency and computer response time.

C. Liu and K. Tomsovic [7] used a production rule language called OPS5 to develop an expert system to assist in the decision-making of the reactive power/voltage control problem (in 1986). The developed system was capable of performing the detection of voltage problems and the application of empirical knowledge to come up with appropriate control actions. Empirical rules for reactive power voltage control were identified first, then they were justified theoretically. Based on the identified knowledge, a set of production rules were built. Numerical examples based on a modified IEEE-30 bus system were presented. Different scenarios of the system condition were selected to test the capability of the developed expert system. The results were very encouraging.

D.J. Sobajic and Y. Pao [12] demonstrated, in 1988, the feasibility of using a knowledge-based expert system for power system contingency screening. Both single and multiple line outages were addressed in the scheme. The expert system was designed to effectively (1) detect and screen out harmless contingencies, and (2) recognize potentially harmful contingencies and determine corresponding endangered areas. The rules incorporated into the knowledge base system were based partly on human operator experience, and partly on power system simulation methods. Results clearly showed such systems can be very effective in focusing attention on a much-reduced number of the more critical contingencies. The expert system was demonstrated using a power system model of moderate size (25 buses, 35 lines).

R.D. Christie and S.N. Talukdar [13] presented , in 1988, a preliminary design of an expert system for on-line security assessment. The paper argues that automatic security assessment scheme, cannot generate information of high quality until their architectures, which are wholly algorithmic now, are changed to hybrids combining algorithms with expert systems. In particular, numerical algorithms should continue to be used for simulating the effects of contingencies, but expert systems should be developed for selecting contingencies and interpreting the simulation results. The authors discussed some of the problems associated with integrating hybrid software into existing Energy Management Systems (EMS) and suggested a solution; namely, the use of a network of work stations tied to existing EMS computers.

In 1988 C. Liu, S.J. Lee, and S.S. Venkata [14] presented an expert system to be used for the restoration of distribution systems. The developed ES will assist dispatchers in locating faults and restoring customers when power lines are equipped with automatic protective switches. It can be utilized as an on-line aid to system operators in a distribution SCADA environment. A knowledge base was constructed of approximately 180 rules and implemented in PROLOG. The incorporated knowledge was acquired from the literature and discussions with distribution engineers. The developed rules were general or portable in the sense that they do not depend on any particular system. The used method was a generalization of the procedures and priority schemes which allows restoration of feeder zones in groups. The results obtained using a prototype system indicated that for a typical scenario, the expert system could formulate a feasible restoration plan within a few seconds.

Alarm message processing seems to lend itself beautifully to ES application. The use of Artificial Intelligence to process alarm messages is currently under active development in several organizations [15].

In 1986, B.F. Wollenberg [1] demonstrated the feasibility of using a real-time expert system to build an Intelligent Alarm Processor (IAP). Few experiments were made to assess the feasibility study, where a prototype of the real time expert system was constructed. The prototype consisted of the knowledge base and already built inference engine program. The language of choice was LISP. A use was made of a dispatcher training simulator (DTS) to generate alarm messages from a sample power system. The knowledge base was obtained by studying the alarm messages from the DTS and then asking, for each message: What would make sense to display and what should be suppressed? " The knowledge base for the IAP treats the alarms as being in one of the following categories; breaker alarms, generation alarms, and line/transformer status alarms. The rules themselves were categorized as follows:

- Alarm level rules -- determine the type of system element for which some parameter is in alarm.
- Generation loss rules -- establish whether generation has been lost through looking at specific elements values and/or statistics.
- Suppress alarm rules -- determine when to suppress an alarm.
- Print alarm rules -- determine when to print (or display) alarms.
- Special message rules -- output special messages derived from combination of different conditions.

In spite of the fact that the rules were not written to represent a final design of a possible IAP, the work so far had demonstrated the feasibility of using AI techniques to solve a very difficult problem facing the operators of power control centers, namely alarm processing.

S.L. Larsen [4] presented a review of a research project that was initiated in 1985 and sponsored by Northern States Power Company's (NSP). The research investigated the feasibility of developing an expert system that would closely emulate the process that an operator uses when evaluating a disturbance. The problems of alarm processing were reviewed first. The author then, highlighted the initial design objectives, which were: keeping the operator aware of the most urgent matters, keeping the operator aware of the problems as they occur, reducing alarm loading and presenting the strategic situation, deeper analysis of situation, predicting near future conditions and finally reminding the operator of actions to be taken. The design would give operators three types of alarm summaries. These include standard alarm, chronological, and priority summaries.

In 1988, H. Admink, et. al. [5] reviewed some of the alarm processing problems in modern power control centers. Two techniques, statically adaptive and dynamically adaptive message processing, were introduced. The concept of statically adaptive message processing included message routing, prioritizing, acknowledging and segmenting. It applied regardless of the state of the power system. While the concept of dynamically adaptive message processing was intended to alleviate the added message burdens encountered during power system disturbance. This was accomplished through automatically adjusting to sudden and unexpected requirements by altering the priority levels. Yet, the problem of reducing a number of discrete messages to a single succinct statement of a condition is one that has not been solved through adaptive message processing.

Application of AI techniques to alarm processing seems to be promising approach. As a drawback, AI has the potential of requiring larger data bases, faster data acquisition systems and extensive computer resources albeit for short duration only.

The following observations can be drawn from the above literature survey:

1. The expert systems application to power system operations, in general, and to alarm processing in particular is a very promising method to assist the power system operators especially during power system disturbances. Yet, this field is still under research by different electric utilities and universities.

2. The construction of any expert system should, in general, go through the following development phases:

a. Defining the problem, clearly and thoroughly.
b. Acquiring Knowledge -- from experienced personnel and literature.
c. Specifying interfaces -- if any is required.

d. Selecting the tool or language -- Should be practical and satisfies the desired application requirements.

d. Building the knowledge base.

e. Developing a prototype system.

f. Expanding the prototype to a practical application level.

1.3 PROBLEM DESCRIPTION

Current power control centers can scan several times as fast as it could 10 years ago. Today it is not uncommon for power control centers to monitor 20,000 to 50,000 points. In a typical power control center an alarm processor program is employed to handle the alarms which are detected by the data acquisition subsystem. Alarms are processed and given to the operator on CRT displays very rapidly. The improvement in the control center's hardware and software today, makes it possible for alarm processors to produce five hundred (500) or more alarms per minute, and this has led to concern about the way alarms are processed [1,5].

When a major disruption occurs on the power system, operators can be overloaded with alarm messages (several hundreds per minute). Because many of the alarm messages are redundant or present information related to the same event, the operators may have difficulty in understanding precisely what has happened. A very essential and important question is whether the operator can analyze and decipher the alarm data into usable information in the time frame which will allow him to react and limit the consequences of a disturbance. It is almost impossible that the operator can take best actions or sometimes the appropriate action when massive numbers of alarms are displayed at such rates.

Standard methods of ranking and classifying alarms were not considered adequate solution to the problem because the importance of alarms depends on the disturbance and changes with condition as disturbance proceeds. Further, they do not provide concise and clear diagnosis as what has happened on the network.

A further complication to the problem is envisioned when false alarms are displayed or when alarms are not issued where they should. This behavior can result from malfunction in the field instrumentation i.e., a case can be a line is out of service and the Mega Watt meter still shows a flow of MW on the line.

The diagnosis of what is happening on the system needs to be made using all of the alarm messages available to insure reliability. Further, the diagnosis should be made quickly so that the operator is informed as soon as possible of the condition. To carry out such a diagnosis of system events requires techniques which involve processing of logical and symbolic information in addition to some of the usual numeric data available at the EMS. The status of a system can be stored in the computer by a description consisting of a list of symbols (words or word phrases) which are known to be true, false or unknown. The process of diagnosis then involves reasoning by manipulating these symbols. The techniques to do this are called EXPERT SYSTEMS and have emerged from research groups in recent years and are now being applied to a variety of problems.

The main objective of this thesis is to develop an expert system that will utilize the alarm messages together with the real time data base of the Energy Management System (EMS) to provide the operators with an analysis of the state of the power system. If, for example, the power system experiences the loss of a transmission line, the EMS may display up to ten or more breaker status changes together with line disconnection and de-energization messages and perhaps some low voltage , Mega watts and Mega Vars messages. This may take more than a full CRT screen to display and certainly cannot be analyzed in detail very quickly by an operator. What is needed is an analysis of the alarm messages to determine that the line has been switched out and display of that information only. All other messages are redundant (in this instant) and can be suppressed.

1.4 SCOPE OF WORK

The primary objective of this thesis is to develop an expert system that reduces the number of power system alarms received at the power control center (PCC) when the network experiences a disturbance or an abnormality. The received alarms will be reduced into fewer, more concise and intelligent messages describing what has happened on the network. All other related and redundant alarms which the dispatcher does not need to see, will be suppressed. The received alarms will be the main input data to the ES and the produced intelligent messages will be the output of the ES to the PCC terminals. The developed ES should emulate an experienced dispatcher's way in analyzing received alarms to reach a conclusion about the state of the network after the occurrence of new events.

The following tasks describe the different steps of the scope of work, in order:

1. Establishment of an overall understanding of the domain. This is accomplished via two major sources:

a. Literature review of the alarms processing, generation and classification in modern power control centers. Different point types, classes, and categories monitored by PCC, which represent the values or the states of power system components, will be studied.

b. Questioning Experts in power control centers. A questionnaire will be constructed and distributed to several power system dispatchers in a Power Control Center . The questionnaire requests dispatchers to specify the power system disturbances that cause burst of alarms when occurred. It also inquires information on how they handle the numerous alarms in case of emergency on the network and what the main indications of the occurrence of a particular emergency are. Such a knowledge can only be acquired from experts in the field and is not usually available in standard texts.

2. <u>Selection of the proper tool.</u> Undergo an investigation of the different available AI languages and tools that will have the following characteristics:

- Applicable and Efficient for on-line applications.
- Rules can be added and deleted without the need to restructure the whole program.
- Can be loaded to a PC.
- Friendly and easy to use.

Tools that possess the above characteristics will be more suitable for an on-line application, particularly, where rules may be altered or expanded occasionally.

3. Identification of Empirical Rules: Utilizing the information gathered in steps 1 and 2 above, the power system alarms will be grouped into sets of related alarms. The observation of each set is a declaration for the PCC of the occurrence of a particular disturbance on the power system. Other files and information that PCC operator views or checks to confirm the occurrence of the event will also be identified and taken into account.

4. Identify Needed Interfaces and Inputs: Identify all information and inputs that are available within the PCC software and which the IAP will need to interface with or access during the on-line operation, i.e the data base connectivity tables. The requirements of these information will be to check or confirm the occurrence of a particular disturbance on the system.

5. Definition of Logical Charts and/or Formulas : After formulating the empirical rules and defining the needed inputs and interfaces, logical charts and equations can be established. These define the decision-making process for each power system disturbance under consideration in order to decide what have happened. Limitations, constraints and assumptions on received alarms and needed data will be defined also.

6. Building the Knowledge Base (KB): The KB will be constructed from the logical relations, derived previously. A prototype knowledge representation of

a small power system network (8-Bus network) will be constructed first. Scenarios for disturbances on this network will be modeled and the developed IAP will be tested for all possible disturbances and malfunction cases. After debugging and successful operation, a bigger system will be selected and the IAP will be tested again. All scenarios representing different power system disturbances will be demonstrated again to verify the capabilities and generalities of the developed expert system.

7. Optimization: The developed IAP will be tuned up to have the following characteristics:

- The developed rules should be general or portable in the sense that they do not depend on any particular system.

- The IAP should be still able to recognize a power system disturbance occurrence from the received input alarms even if some of the alarms that should have been produced for a particular disturbance were missing, due to bad telemetry operation in the substation. Further, the IAP should display a special message indicating probable bad telemetry point in that substation.

- Since the IAP will be implemented in an on-line environment, basically, IAP execution time must be minimized as much as possible.

8. <u>Testing</u>: The IAP capabilities will finally be demonstrated on a large system. Several scenarios will be assumed, each will represent one or more disturbances. The IAP should be able to recognize the disturbances, given only the generated alarms as an input.

CHAPTER 2

EXPERT SYSTEMS IN POWER SYSTEM OPERATIONS

This chapter is divided into three parts. The first part provides the reader with some back ground about expert systems in general. More emphasis is given on knowledge engineering and programming languages and tools since they constitute the main work done in expert system part of this thesis.

The second part highlights topics related to the Energy Management Systems (EMS) in Power Control Centers (PCC) and the associated operational problems. Finally, an overview of the alarm types, classifications, processing and problems in PCC is given.

2.1 EXPERT SYSTEMS COMPONENTS

Expert systems (ESs) are one branch of the field of artificial intelligence (AI) that have the ability, if programmed properly, to emulate the reasoning processes of human experts, in a particular domain, when a problem is encountered. They are used to perform a variety of extremely complicated tasks that in the past could be performed by only a limited number of highly trained human experts [11,16].

2.1.1 The Architecture of Expert Systems

In spite that there is no one common structure for all expert systems, most of the architectures have several general components in common. Figure 1 depicts a general architecture of an expert system. Following is a brief discussion on each component.

Knowledge Base (KB) : The knowledge base represents a storehouse of the knowledge primitives (i.e, basic facts, procedural rules, and heuristics) available at the system [16]. Such a knowledge is obtained from experts whose judgement and knowledge are to be emulated by the expert system and used to make decisions and to determine how the expert system is to treat the input data. Variety of schemes are used for storing information in the knowledge base. The design of the knowledge representation schemes impacts the design of the inference engine and the overall efficiency of the system. A summary of the most known knowledge representation schemes will be presented in section 2.1.2.

Inference Engine (IE) : It is that part of the expert system which

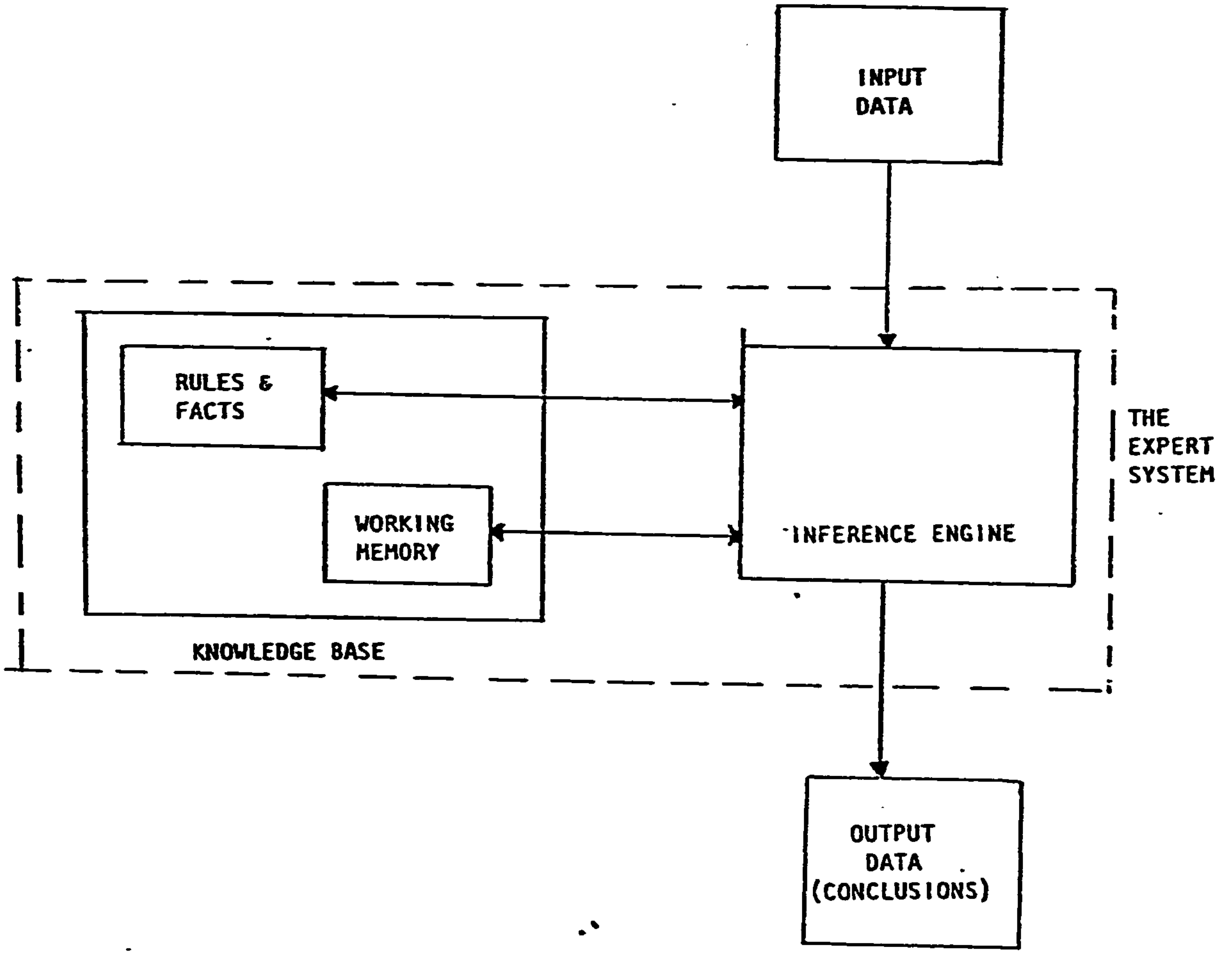


Figure 1: A general architecture of an expert system

manipulates the rules and draws conclusions from the inputs [1,17]. The expert system manipulates the rules in one of two ways:

1. Backward chaining - A type of reasoning that begins at a specified goal and works backward in an attempt to prove that the goal is true. Similarly, it starts with a given problem and works backward to find all possible causes.

2. Forward chaining - Another type of reasoning that begins with the known facts and works forward, trying to find a successful goal from known facts.

Working Memory: It is often called " short-term " memory. It is that component of an expert system where all the input data of the problem currently being examined are stored.

Input / Output Facility: Often viewed as the user interface facility. It must accept information from the user and translate it into a form acceptable to the remainder of the system. Alternatively, it can accept information from the system and convert it to a form that can be understood by the user. The user can be external software files on the system, typically in an on-line environment.

2.1.2 Knowledge Engineering

Knowledge engineering is the process of acquiring specific-domain knowledge and building it into the knowledge base. Although knowledge can be secured from a variety of sources, including documentation and existing computer information systems, most of it must be elicited from human experts. The knowledge provided by the expert will be generally in the form that is oriented toward the subject domain. E. D. Feigenbaum [18] stated, knowledge engineering deals with topics of knowledge acquisition, knowledge representation, and reasoning ability employed during inference.

A Knowledge Engineer (KE) is the person who acquires the knowledge from the domain expert and transports it to the knowledge base. He acts as an interpreter between the expert, extracting knowledge from him, and transforming that knowledge into a program the computer can understand.

In the remaining part of this section knowledge acquisition and knowledge representation will be viewed.

<u>Knowledge Acquisition:</u> Knowledge acquisition is frequently the most difficult and time consuming aspect of ES development. This process

requires locating, collecting, relating, and refining knowledge. This is specially difficult during the development of a system, because experts tend to forget some of their knowledge. Further, it requires extended human communications between the domain expert and the knowledge engineer [16]. Due to the difficulties encountered in the knowledge acquisition, emphasis was made on the design of representational schemes that allow for incremental construction of the knowledge base.

Knowledge Representation: Knowledge representation is viewed as the most active topic in the knowledge engineering. Knowledge, generally, takes different shapes. There is a knowledge about objects, a knowledge about events, a knowledge about procedures, and a knowledge about what we know (knowledge itself), or often called Meta-Knowledge. These different types demand different methods of representation. Many knowledge representational schemes have been devised in the last two decades. Following is a brief introduction of the most known schemes [19].

A. Logic: is one of the first representation schemes used in AI. The inference rule in logic allows the deduction of new facts from previously given facts. A typical rule in a logic form is as follows [22]:

$$(x \ (x \rightarrow y)) \rightarrow y$$

The rule states that if the two statements (x) and (x -> (implies) y) are true, then, we conclude that the statement (y) is also true. Representation in logic was employed in AI for many reasons. One reason is that logic is a natural way of thinking and a lot of knowledge can be expressed in this way. In addition, representation in logic is precise, flexible, and most of all modular; that is the rule can be added or deleted without affecting other rules.

B. Procedural Representation: is another method of knowledge representation and was used in the development of several expert systems like SIR and PLANNER. One advantage of using this technique is that it could easily represent heuristic knowledge.

C. Semantic Network (or SEMANTICS): It consists of:

(1) a data structure of NODES, representing objectives, concepts, situations, and LINKS which explain relations among nodes.

(2) a specialized inference procedure that operates on the data structure.

As the case with the procedural representation, SEMANTICS method is not modular; yet, it is easy to construct [16].

D. Production System: a production system, or often called rule-based system. It is considered the most successful and commonly used scheme in ESs. It uses rules for knowledge representation [16]. Each rule consists of two parts in the following form:

IF (set of conditions)
THEN (actions to be taken)

The "set of conditions" portion of the rule, called the antecedents or preconditions, describes the conditions that must be met for the rule to be applied or executed (most often referred to the term "applied" as "fired" in AI). The "actions to be taken" portion, called consequents or results, describes the actions that occur as a result of the rule's application.

A production system consists of:

- A working memory that is used to track the current state of the universe under consideration.
- A set of production rules (condition- action pairs).
- An interpreter that examines the current state and executes applicable production rules.

One common production system programming language, OPS5 refers to the first two items as working memory and production memory, respectively.

The main advantages of this method is that it employs the same line of reasoning human follows, highly modular and, moreover, it could be very efficient in narrowing down the search space needed to obtain a solution. This is achieved using what is known as METE-RULES. Additional examples of production systems include YAPS, OPS83 and CLIPS.

2.2 EXPERT SYSTEMS PROGRAMMING LANGUAGES AND TOOLS

An Expert System can be constructed with a variety of software tools such as LISP, PROLOG, and special expert system development environments, called "shells". Since the detailed discussion of AI languages and tools is beyond the scope of this thesis, a brief discussion of the most two common languages, LISP and PROLOG, and a highlight on some tools employed in the design of expert systems are given in this section.

To illustrate the principle ideas of expert systems, a comparison between them and conventional programs is made. There are several major differences between the two types of programming techniques.

First, Expert Systems are designed to manipulate symbolically expressed problems while many conventional programs primarily handle problems that are numerically expressed. Second, reasoning and deductions in expert programs rely "heavily" on heuristics. Conventional programs, on the other hand, depend mostly on algorithms. Third, programmers are in charge of maintaining and modifying conventional programs while experts and knowledge engineers, individuals who design expert programs, maintain and monitor the program's performance. Finally, one very unique and important characteristic of expert systems is the separation between the domain knowledge, "Knowledge Base", information about problem being solved, "Input Data", and the method used to solve the problem, namely the "Inference Engine". This feature of expert systems simplifies the modification process of such systems. Such programs can be evaluated and then updated by modifying its "knowledge base". In contrast, to update a conventional program, the whole structure may have to be altered [17,19]. Some of the difference emphasis between the two methods are illustrated in Table A.1 in Appendix - A..

LISP which stands for LISt Processor, has been considered one of the most commonly used AI languages for expert systems design. LISP is a language designed for general symbolic manipulation. It works with symbols and gives meaning to a symbol by using it to represent a fact or an action whose result is a piece of data. The manipulation of lists enables a programmer to simulate human cognitive ability, which often involves going through ordered lists of actions, crossing them one by one until the end of the list is reached [11,19].

PROLOG is another popular language, which stands for PROgramming in LOGic. It is a higher level language for dealing directly with objects and relationships. A PROLOG user can submit facts and rules and then consult the resultant data base for solutions to various problems. A program written in PROLOG is mainly a descriptive database of facts and rules that can be executed to solve a problem. Facts are represented in a predicate-logic style and can be added and deleted very easily. PROLOG gained a lot of momentum when it was chosen by the Japanese as the language of their Fifth - Generation computer project [11].

With the increasing applications of AI, many efficient expert system shells were developed that can serve the same purpose of developing an Expert System from scratch but saves a lot of time and effort. Typically these shells consist of [19]:

1. The reasoning mechanism (inference engine) that uses knowledge to form conclusions and answers.
2. Knowledge - base structure.
3. The linguistic module (usually PROLOG or LISP) needed to operate the program.

The expert system developer only needs to add the knowledge about the problem in the form of rules, facts and data, to have a working system. This means that an expert system can be developed much faster with a shell compared to the several months or years necessary to develop a working system from scratch using either LISP or PROLOG [11]. Examples of such tools are EXPERT, KES, OPS5, KEE, CLIPS, and few others. Table A.2, in Appendix - A, gives some of the properties of these tools. CLIPS is chosen for building the expert system in this thesis. The major features of it are highlighted in the next section.

2.3 CLIPS

A lot of effort was spent on selecting an Expert System shell which will satisfy the requirements needed for this application. The major criteria used for selecting the tool are: easy to interface with other languages, modularity (can add more rules without the need for reconstructing the whole program), can be installed on IBM PC and most important is affordable.

The basic requirements needed in the Expert System to satisfy the thesis application are:-

1. A run-time module (or an executable version) can be created out of the developed expert system. This requirement is very essential for an on-line batch processing, typically, this is the case in EMS environment.
2. Allows for batch processing.

3. The consultation paradigm (pattern) of the language should be diagnostic and prescriptive in nature.

Studying the characteristics of several languages indicated that CLIPS is an appropriate tool.

CLIPS (which stands for the 'C' Language Production System) provides reasonable performance on a wide variety of computers. CLIPS was developed by the artificial intelligence section (AIS) at NASA/Johnson Space Center. It is a rule language based on the Rete algorithm. The Rete algorithm was specifically designed to provide very efficient pattern matching. CLIPS has attempted to import this algorithm in a manner that combines efficient performance with powerful features. When used properly, CLIPS can provide very reasonable performance even on microcomputers. However, to use CLIPS properly requires some understanding of how the pattern matcher works.

The CLIPS expert system development tool consists basically of facts and rules. Facts are data items that drive CLIPS as a FORWARD CHAINING expert system.

<u>FACTS</u>: consist of fields which may be in the form of:

1. numbers (decimal or integer)

2. words (must start with an alphabetic character)
3. strings (in quotation marks)

Facts are introduced to the system by two ways:

1. Defining a single fact: The syntax is (assert fact)

i.e. (assert (Friday is a holiday))

Where the fact is " Friday is a holiday", assert is the defining word for facts in CLIPS.

2. Defining multiple facts: The syntax is:

```
(deffacts    fact 1 "comment(s)
             fact 2
             fact n))
```

Where deffacts is the key word declaring new facts to be added and it stands for define facts, and "comments" is an optional feature for adding facts, if needed. All comments will not be executed as far as they are within double quotes.

Facts may be:

1. Asserted (added) before execution.
2. Asserted (added) as the action of a rule.
3. Retracted (removed) as the action of a rule.

Facts are placed in a fact-list and stored in the working memory (agenda).

<u>RULES:</u> As it is the case with all production rules in ES, rules consist of two parts: the IF part, called the antecedent or premise, and the THEN part called the consequent or conclusion. Rules are driven by facts and can only fire when the IF part of the rule is matched by the facts in the agenda. Rules are introduced to the system in CLIPS in the following manner:-

```
        Syntax: (defrule.. rule name .."comment(s)"
                    (pattern)
         or         (several patterns)
        =>
                    (action)
         or         (several actions))
i.e.    (defrule  warning " this is a sample rule in CLIPS"
        (student get F)
        (student GPA below 2.0)
        (student has less that 12 units)
      =>


        ( fprintout t "student deserves a warning" crlf))
```

where a warning will be printed in this example if student meet the first three conditions.

The premise (LHS, Left Hand Side) of a rule comprises one or more patterns. The conclusion (RHS, right hand side) of a rule comprises one or more actions. It is worth noting that it is not required to type IF and THEN in CLIPS, rather the system will assume that statements before the arrow (=>) are the IF part and the ones after it are the THEN part.

In addition, all defining statements, for both rules and facts, should be placed between brackets.

Prior to initiating execution, each rule is loaded into the system and a network of all patterns that appear on the LHS of any rule is constructed. As facts are asserted into the fact-list, the facts are filtered through the pattern network. If the form of the pattern (number of fields and literal fields) matches any of the patterns in the network, the rule(s) with that pattern is partially instantiated. When facts exist that match all the patterns on the LHS of the rule, variable bindings (if any) are considered. They are considered from the top to the bottom; i.e., the first pattern on the LHS of the rule is considered, then the second, and so on. If the field values for all patterns are consistent with the constraints applied to the variables, the rule(s)is activated and placed on the agenda.

CLIPS provides reasonable performance on a wide variety of computers. It was specifically designed for portability and has been installed on several kinds of computers without modifications to the source code (these kinds include IBM PC, Honeywell, Hewlett Packard (HP), VAX/VMS, etc.). Further, it should run in any system which support a full Kernighan and Ritchie 'C' compiler. It will also run an any 'C' compiler compatible with the proposed ANSI standard.

Even though CLIPS was developed in 'C' language, it can be embedded within a program written in another language. In addition, a run time module can be created that compile all the KB rules into a single executable. Such a transformation reduces the size of the executable image.

Dynamic memory is required for the rules, facts, and internal structure during execution. To optimize both storage and processing speed, CLIPS does much of its own memory management.

As a drawback, a problem that is often encountered involves linking to inappropriate system libraries. For example, when using a compiler which supports different memory models, the user must link with the system the libraries that match the memory model that the CLIPS model was compiled under. The same can be said for the floating point models. Some computers provide multiple ways of storing floating point numbers (typically differing in accuracy or speed of processing). The system libraries which use the same storage formats that CLIPS code was compiled under must be linked with the system. More details about the language and its performance can be found in Reference [20].

2.4 EMS OPERATION: THE DESIRE FOR IMPROVED SOFTWARE

Modern power systems are operated by highly skilled operators through computerized control systems. The Energy Management System (EMS) is the center of a control system organized in a hierarchical structure utilizing remote terminal units (RTU), communication links, and various levels of computer processing systems. The function of EMS is to ensure the secure and economic operation of the power system as well as to facilitate the minute-by-minute tasks carried out by the operations personnel. The EMS is mainly designed to be used in the "normal" state where such functions as state estimation, security analysis, and optimal power flow are used to ensure secure operation. On the other hand, functions such as automatic generation control, economic dispatch, unit commitment, and load forecasting are used to ensure that the most economic operation is obtained. Much of what happens in normal operation is now computerized and human operators only intervene to carry out the few manual tasks required. The picture is quite different, however, when we look at the use of an EMS during an unforeseen event or a failure of major components on the power system. In such instances, the EMS serves mainly as an information gathering and reporting system and the sophisticated application software that function in normal operation may be of little use. For example, in a regulatory shutdown of all nuclear units, human operators will take over the economic dispatch and unit commitment to reschedule economic operation. Similarly, when a sudden loss of transmission equipment occurs it is human operators who must understand what has happened and decide on what actions to take. It is especially during such emergencies that conventional software is less effective. The requirement for smarter software thus becomes more important in such instances [11].

Coping with emergency events is referred to as a diagnosis and decision process. The solution of such process rest heavily on the experience and skill of the human operators to react correctly. Moreover, power systems organized in an hierarchical form have become very complex because of structure, status and relevant technical issues. This growing complexity is causing problems. In addition to the present power system operation problems discussed above, some more are summarized:

(1) A rapid increase in the number of real-time messages has made operator response more difficult. This difficulty, called "human cognitive barrier" must be overcome.

(2) Current numerical processing software cannot meet the operational requirements of power system in some situations. Examples are processing during emergency conditions and using software in situations beyond their design limitation.

(3) Most design, planning, and control problems encountered are complex and time consuming because of multiple objective functions, multiple

constraints, complex system interactions, the need for trade off, and so on [2].

2.5 ALARMS IN POWER CONTROL CENTERS

All modern Power Control Centers have some form of alarm processing to alert the power system dispatchers to power system parameters that are out of normal range or to undesired changes that may affect the operation of the power system.

Current Power Control Centers (PCC's) can scan several times as fast as it could 10 years ago. The sophistication in the today's hardware and software had made it possible for the PCC's to monitor 20,000 to 50,000 points. Ironically, in cases of really serious and complex situations the operator would be flooded with many alarms and a lot of information which tend to hamper or confuse rather than assist his decision-making process [5].

In a typical control center alarm processor program is employed to handle the alarms which are detected by the data acquisition subsystem•Alarms are processed and given to the operator on CRT displays very rapidly. The improvement in the control center's hardware and software today, make it possible for alarm processors to produce five hundred (500) or more alarms per minute, and this has led to concern about the way alarms are processed.

In this section the data acquisition subsystem in the power control center will be discussed briefly, then a brief discussion of the alarms generation, conditions, and processing will follow. Finally, the need for AI application in alarms processing will be highlighted.

2.5.1 Data Acquisition Subsystem (DACS)

The Data Acquisition together with the Communications form a subsystem that is referred to as DACS in the modern control centers. The DACS has as its prime function the transfer of current state of the electric system from the field to a digitized data base in a control center computer [21]. Through use of that data base by Man-Machine and Applications Software, the control center operators are able to monitor and control the electric system according to company established operating procedures. The control center may be designed for energy control (or often called Energy Management System EMS), Supervisory Control And Data Acquisition 'SCADA' functions, or a combination of the two. SCADA center provides very typical functions of monitoring, logging, and supervisory control. The EMS Center typically provides for the functions of generation control, security analysis, study and logging applications and may also provide the SCADA functions.

The relationship of the DACS to the EMS center and to the electric system can be visualized as in Figure 2. The DACS is the interface, the data and control path to the generating plants and substations equipment, to regional control centers, to neighboring utility control centers, and to a power pool centers. In short, it is the interface to the external world [21].

While there are many possible designs and configurations for a Data Acquisition Subsystem, they all have certain basic equipments in common. Namely; Communication Interfaces, Modems, Remote Terminal Units (RTU's), and the DAC Software in the PCC. The DAC software scans the different RTU's periodically requesting data to support the operational needs of the other software in the control center. It will process all received data and create a database which presents a digitized image of the electric power system that is accessible by the other software. The DAC software will also process requests from all other software for transmission of supervisory and generation control commands to remote terminals.

2.5.2 Control Centers Data Point Types

The PCC scans the different elements of the power system network, that is interfaced with the RTU, periodically to update the digitized image of the power system network in the computer system. The scanning is typically performed every

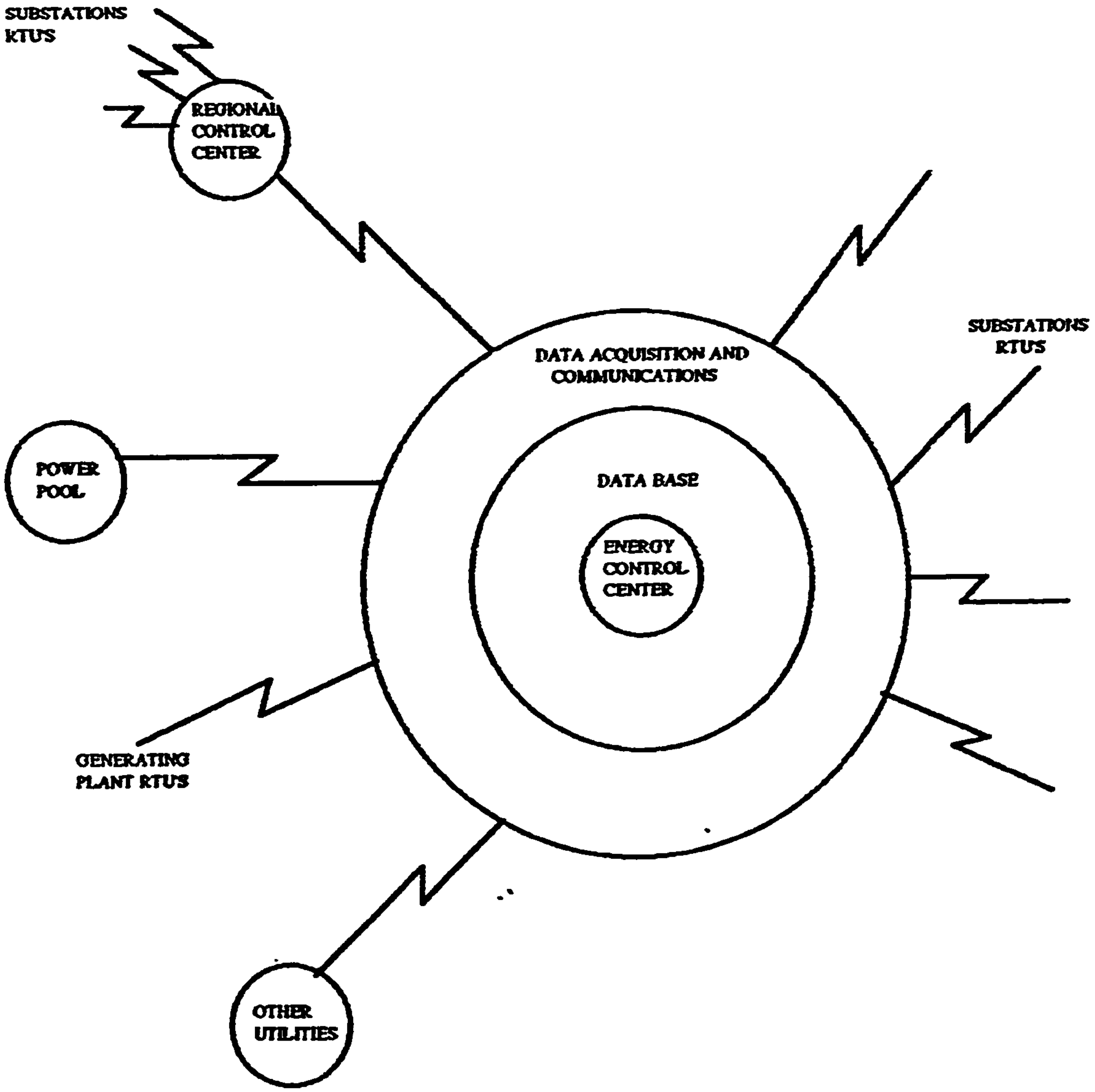


Figure 2: The relationship of the DACS to the EMS center and to the electric system

2-10 seconds. The acquired data can be either one of the following major point types [22]:-

1. Digital (status) input - A point that has one or more discrete states and may be used for alarm, indication, sequence of events, or device status. An example of this type is breaker status, trip or close.
2. Analog input - A point that is represented by an available analog signal. May be used for voltage, current, MW, and other analog measurements.
3. Accumulator input - A point that is an accumulating or counter type measuring device. Kilowatt-hour reading is a typical accumulator point.
4. Control output - Interposer relays that are actuated from the control center to operate field devices such as circuit breakers.
5. Analog output - A set-point or desired value to be used by a local device controller. Some generation control approaches send desired generation to the generator control unit as an analog set-point.

2.5.3 Alarms Types and Classifications

Alarms are generated for a variety of system conditions among which are: circuit breaker changes, current or MW limits exceeded, frequency deviation, voltage deviation, operation of protective equipment, non-intervention of remote control communication errors, etc. [1].

Alarms received by the operator in the power control center can be classified in different ways. One way is:

1. Power system alarms: Alarms that reflect changes and/or abnormality in the power system network. For instant, breaker alarms, line/transformer status alarms (line disconnected or open at one end), return to normal alarms, generation alarms (generator off line), line/transformer loading alarms, bus voltage alarms, frequency alarms, etc.
2. Field device failure alarms: These alarms reflect failure of devices, in the station, which are part of the remote control circuitry, i.e. A/D converter, transducer failure, loss of DC power in the remote terminal unit, etc....
3. Communication alarms: These alarms are generated for either loss of communication, detecting some communication equipment breakdown, or bad communication circuit (high loss or a lot of noise).
4. EMS alarms: When EMS programs detect a problem or bad value reading, i.e. whenever the bus voltage value calculated by the state estimator program does not agree with the telemetered value, an alarm message is sent to the operator.

5. Control system alarms: Alarms that are generated when one of the control center components fail or undergo a software or a hardware problem [22,23].

2.5.4 Alarm Generation

Implementation of alarm generation consists of two distinct tasks: First, the detection and identification of each alarm condition. Second, the display of sufficient information in the form of a short CRT message to enable the operator to understand the alarm condition and take the corrective action [22].

The detection and identification task (item 1) is implemented by software routine distributed throughout the DAC subsystem and application programs (EMS), where the data for the alarm condition initially exists. It is the responsibility of these programs to detect and identify alarm conditions and also to determine when the alarm condition has been properly corrected and no longer exists.

The alarm message generation program assembles a short and concise alarm message and initiates its display (item 2) to the operator on his alarm CRT. An alarm message may contain the following type of information: time, date, text message, equipment name and location from the data base directory, and dynamic data values and status from data base and limit values. Figure 3 displays examples of some alarm cases. The first alarm was generated for an over-voltage on a line '

```
Time      Loc.  Eq.   I.D.  Text          lim. val.  Unit    Cur.Val
--------------------------------------------------------------------
06:00:00  SS 1 Line   L4D   Exceed limit     230      KV      245

06:10:05  SS 2 BRKR   34A   Trip          Normally close

06:11:12  SS 5 Line   LD3   Loss of relay set 1

07:12:13  SS 3 Trans  TR1   MW drop to value  100    MW      .2
--------------------------------------------------------------------
```

Figure 3. Typical Alarm display format

LD4 '. The second for a breaker changing status. The third for loss of relay set on line 'LD3'. Finally, the last alarm for drop of MW on transformer 'TR1' to zero value.

The Alarm List is the current list of existing alarm messages, which is stored in fast memory. A CRT display program needs to be called to display the Alarm list with the new messages added. New messages are added at the bottom line of the Alarm CRT and the existing messages are pushed upward.

2.5.5 Alarm Conditions

Conditions on the power system and equipment monitored by the EMS can cause the activation of large number of alarms. These alarms may bevalid or invalid depending upon circumstances. Some examples of conditions which create burst type alarms are as follows [24]:-

System Start-up -- When an EMS is initialized or when it is restarted from a failure condition, all temporary data that does not match a "default" (start up) condition will appear to the EMS as abnormal situations. As a result, alarms are issued. This will occur during the initial system scans thus causing hundreds or thousands of alarms to be generated in a burst mode.

Data Set Switch -- An EMS requires constant additions and corrections to the data base in order to properly reflect the equipment in the field. Most of the time, the corrections will be minor and will not require a regeneration of the data base. The corrections will be made in an off-line data set and will be switched to replace the on-line data set. Status points in the new data set which differ from the scan data status will generate alarms.

Communication System Dropout -- The backbone of any EMS is the communication system which ties the system computers with remote terminal units (RTUs). A disturbance which disables communications can cause large numbers of communication errors in addition to a backlog of alarms and state changes that will be detected when the disturbance has cleared.

System Disturbances -- As storm and weather fronts move across the power system, numerous alarms will be received in bursts. The detected alarms can exceed 15 per second. Operators can easily be overloaded with alarms from the portions of the power system being affected by the disturbance.

Emergency Condition -- Extreme condition which threaten the integrity of a power system can cause so many alarms that the system operator could not use the EMS.

2.6 EXPERT SYSTEMS APPLICATION IN POWER CONTROL CENTERS

Research in the area of EMS operation indicates the need for knowledge-base software application in power systems operations. A fundamental motivation for such software is the need to overcome the human cognitive barrier which EMS installation encounter during emergency operation or when application programs are used beyond their design limitations [15].

The cognitive barrier is felt as the complexity of power system operations increases without sufficient efforts to cope with it. This is true of today's EMS installation where the quantity of data gathered and the rate at which they are gathered can overwhelm a human operator.

It must be noted that the driving force in EMS complexity is the desire to operate the power system closer to its limits so as to make better use of generation and transmission facilities. This, in turn, has made a qualitative change in system operations requiring quicker diagnosis and decision making by operators. Figure 4 illustrates this situation. While system complexity increases steadily, the operator's ability to cope with it decreases. Since the complexity of power operation is very likely to continue to increase in the future, there is a risk of human operators being unable to manage certain functions unless their capability is enhanced.

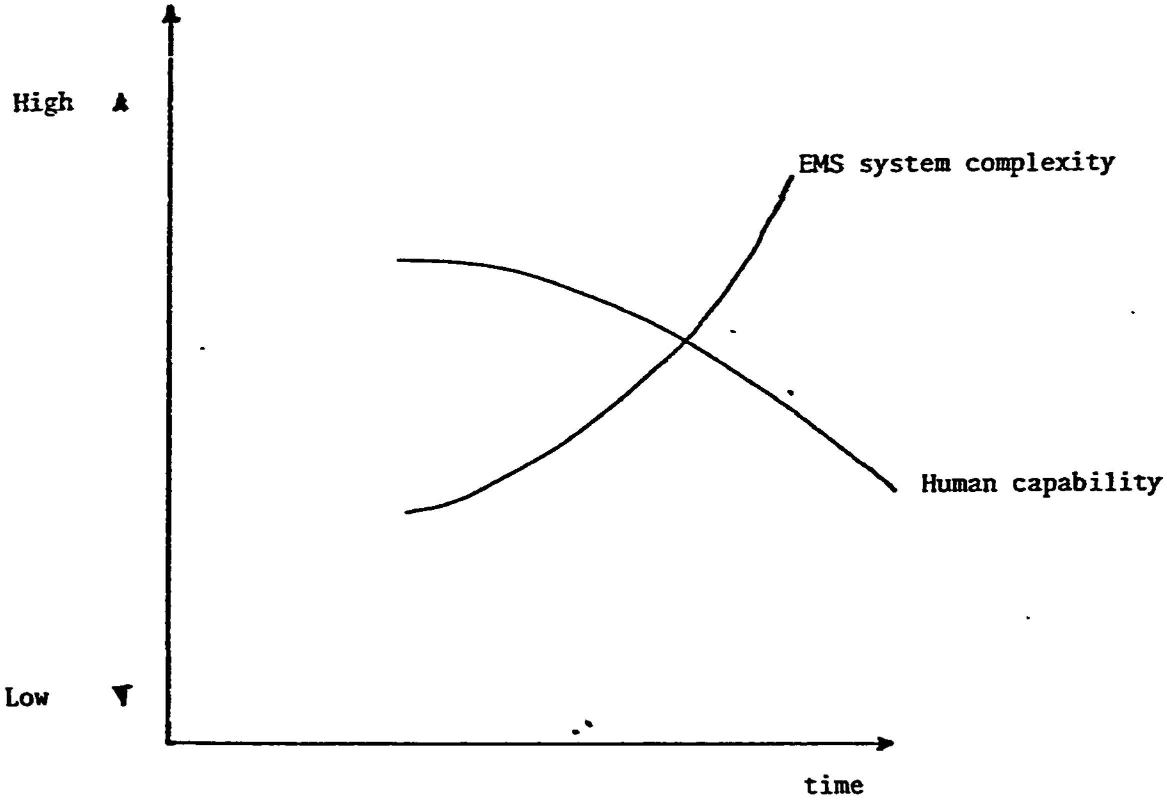


Figure 4: The risk of enlarging the human cognitive barrier

As indicated earlier, the cognitive barrier is quickly realized in power systems operation when sudden and unforeseen events occur.When human operators meet such an event, they have to understand the situation (diagnosis) and determine actions (decisions) to return the system to normal. As all the tasks have to be done in real time, the operators are exposed to heavy mental stress and this makes the cognitive process distinctively different from that experienced by others. For example, experimental designers of large-scale integrated circuits often face similar cognitive barriers but solutions are not required in real time. There are several ways to help operators overcome this cognitive barrier. First, operators need to understand what is happening on the power system and the AI software can give guidance by showing various scenarios that explain the situation consistently. Operators can then check for the most plausible scenarios, some of which may have been overlooked. Similar guidance can be expected in using large application programs where the AI system can guide the operator in its use. Further, since power system operations are filled with many fragmented tasks that are done almost routinely, smart software to do such routine tasks could relieve operators and allow them to devote their time to more important tasks.

Artificial intelligence may be the next major enhancement for the functions of an energy management system. However, implementing AI in an EMS is more difficult than adding a new application program written in an engineering language such as FORTRAN. AI programs are generally written in special languages because of the needs for symbolic processing and in some cases these languages require special hardware.

Most AI development work is currently done in LISP or Prolog. However, it is unclear at this time whether these languages are the best to use in implementation of AI into EMS systems. There are basically two approaches that can be taken [2,4]:-

a) Implement AI programs in languages such as Pascal or C that contain many of the necessary language features needed in AI programming. An alternative way would be to implement the AI programs directly in LISP on conventional EMS hardware. This approach suffers, however, in that such hardware is often not efficient at running programs written in LISP.

b) Attach special hardware to the EMS that runs LISP or Prolog effectively.

The decision is further made difficult by the fact that AI development thrives best in a "Prototyping" environment which may not necessarily be the same environment as the "delivery" environment. Going with the first approach allows the AI programs to interface directly with existing database, display, application and communication software but requires rewriting (or translating) the programs if the development language is different from the delivery language. The second approach eliminates the rewriting or translation but adds the problem of communication between two different hardware components.

2.7 SUGGESTED METHODS FOR IMPROVING ALARMS PROCESSING

The ability of the power systems dispatchers to digest the large volume of alarms and messages reported in a modern control center is limited. Methods for alarm and message processing have not kept pace with the increase in volume of information reported to the EMS/SCADA systems. Several methods were suggested to enhance alarm processing in control centers [5], these methods can be categorized as follows:

First : Statically Adaptive technique - It can provide means for controlling and channelling messages by carefully routing, prioritizing, and segmenting messages where the user may adjust the information to his particular requirements.

Second: Dynamically Adaptive technique - This technique is more valuable during power system disturbances where alarm processing programs are set to automatically adjust to sudden and unexpected requirements. This is achieved by altering priority classes to limit information presentation to only the most essential elements. In addition, an automatic alarm acknowledgment scheme could be activated to relieve the dispatcher of this function.

Three: Implementation of an Intelligent Alarm Processor - Artificial intelligence programming is used to develop an expert system tied to the alarm processing subsystem to reduce the number of alarms displayed on the dispatcher's CRT.

The first two techniques, which were also elaborated in the literature review, were not considered adequate solutions to the problem. This is due to the fact that the degree of importance of alarms depends on the disturbance and changes with system conditions as the disturbance proceeds [5]. Further, the reduction of a number of discrete messages to a single representative message of a condition is still to be solved. The dispatcher needs to know quickly and accurately the " bottom line" of a number of operations which might accompany the system disturbance.

The essence of applying AI into alarm processing lies in the idea that alarms should not simply be rearranged for display. Rather, the alarm messages together with the data in the real time data base of the EMS should be used to provide the operators with a continuous real time analysis of the state of the power system. In effect, the EMS computers should aid the operators by doing much of the analysis of the alarms that would have to be done anyway.

AI developers have suggested several ways to solve some of the EMS problems. In addition, the AI research has spawned new ways to develop software as well as new hardware to efficiently run that software. What is needed now is: first, a development of a real-time expert system to be applied to the PCC alarms. Second, an efficient way to use and embed this software, hardware, and development environment combination into a real time energy management system. In general, AI will have to have the following (among others), to be useful in an energy management system [15]:-

- o High speed execution of the software: Energy management systems being real time systems are very time critical and high speed processors are an absolute requirements to make them operate properly.
- o Coping with existing system facilities: Any AI application will of necessity require coping with the on-line data base, man-machine display system and general operating system utilities of the EMS system.
- o Ease of development and maintenance of the AI software: Many of the development and maintenance tools now used in stand-alone work stations must be made available for the EMS AI software environment.

CHAPTER 3

BUILDING PROCESS OF THE IAP

The general sequential phases of building the Intelligent Alarm Processor (IAP) are:

1. Acquiring the needed knowledge.
2. Establishment of Empirical rules.
3. Building a prototype for a small network.
4. Generalization of the prototype to any power system network.

In this chapter, the four phases involved in building the IAP will be discussed. The structure of the IAP will also be studied and the logic followed in building it will be reviewed.

3.1 KNOWLEDGE ACQUISITION

Several steps were adopted in acquiring the power system alarms information and the way they are processed and analyzed by power dispatchers in power control centers. The first step is to survey available literature in the field of alarms processing in power control centers. This has already been presented in chapter 2. The second step is to acquire additional knowledge from experts. To achieve this task a questionnaire was prepared and distributed to power system dispatchers. It inquires certain information that is not available in the literature and can only be found with experts in this field. Final step was to observe power systems dispatcher reaction during power systems disturbances and gather some cases that will assist in building empirical rules for alarms processing during disturbances.

Knowledge Acquisition From Experts

A questionnaire was prepared and distributed to several dispatchers at the SCECO EAST Power Control Center. It includes the following questions:-

Q1. List the power system disturbances that, when occurring on the network, generate many alarms (i.e. line out of service, loss of load..etc ?)

Q2. For each of the disturbances you have mentioned above, what are the associated alarms that are displayed at the PCC (include all alarms generated for each disturbance) ?

Q3. Suppose that you received many alarms suddenly, what are the first things you look at? Which alarms are the most peculiar and decisive that you will check for their presence first to derive your conclusions?.

Q4. If there are several disturbances on the network, can you rank the priority in which you would like them to be displayed (assume the disturbances that were defined in the first question).

Q5. For each case explained above (in questions 1 & 2) what messages do you think should be displayed and what should be suppressed (not displayed) ?.

The following was concluded from the received responses:-

1. There are certain disturbances which occur frequently and usually generate a huge burst of alarms. These disturbances are summarized in section 3.2 together with their associated indicative alarms. (Indicative alarms are defined as the group of alarms that come up every time a particular disturbance occurs).

2. There are certain indicative alarms that accompany any disturbance. When these alarms appear a conclusion can be made.

3. There are secondary alarms that are generated occasionally upon the occurrence of power system disturbances. These alarms are either one of the following two types:

a) Alarms of no importance or of redundant information to the dispatcher and should be suppressed. Further, these alarms can not be used to judge the occurrence of a disturbance.

b) Alarms that are useful to know and should not be suppressed. These alarms are usually related to the power system in general.

A sample questionnaire which includes the dispatcher's response is shown in Appendix - B.

3.2 Alarms Generated During Major Disturbances

Dispatchers answers to the questionnaire indicated that there are certain major disturbances that occur more frequently on the power network. The occurrence of these causes huge number of alarms to be observed at the PCC. Furthermore, among these alarms, there is a group that will be observed each time a particular disturbance occurs. This group was called indicative alarms since they indicate to the dispatchers the occurrence of a particular disturbance.

3.2.1 Indicative Alarms

The major disturbances together with their indicative alarms are provided below:

Case a: <u>Generator off-line</u>

The following alarms are depicted whenever a generator goes off-line:

- Tripping of all breakers that connect the generator to the bus.
- MWatt reading on the Generator output Bus $< E1$.
- MVAR reading on the Generator output Bus $< E2$.
- Voltage reading on the Generator output Bus $< E3$.
- Frequency reading on the Generator output Bus $< E4$.

where $Ei = a(v) \% * N(i)$

$a(v)$ = accuracy of the telemetry device 'v' in percentage.

$N(i)$ = nominal reading at equipment 'i' (i.e., 60 Hz for frequency).

Where numbers 1, 2, 3 and 4 refer to MW, MVAR, voltage, and frequency respectively.

The above alarms will always occur whenever a generator goes off-line, provided that all the MW, MVAR, KV and Frequency telemetry devices are installed on the Generator output Bus. However, other alarms may be generated as a result of a generator going off-line. This is subject to the ratio of the MWatt lost, as a result of the generator failure, over the total generation at that moment, these alarms may include:

- Voltage drop in that substation and contiguous substations.
- Frequency drop.
- Some MWatt overload on contiguous transmission lines.

Even though these alarms may be related to the generator going off-line, they should not be suppressed. For this case, the dispatcher need to be alerted that a generator went off-line and some trouble has been caused in that area as a consequence. Thus necessary action may need to be taken. This type of alarms will be discussed in later sections.

Another type of alarms may also be generated occasionally. This type includes alarms on relay sets operation, SF6 Gas urgent alarms, oil pressure alarms, etc..These alarms should be suppressed because the dispatcher cares to know more what has happened on the network when a disturbance occur. However, these alarms will be saved on a logger and can be referred to any time.

Case b: Transformer de-energized

Following are the alarms observed when such a disturbance occurs:

- MW reading drop < E1 (on both Low side (LS) and High side (HS) if both sides are telemetered).

- MVAR reading drop to < E2 (on both LS and HS if telemetry is defined on both sides).
- KV reading drop to < E3 (on both LS and HS if telemetry is defined on both sides).
- Tripping of all connected breakers.

Where E1, E2, and E3 are as defined above.

If the transformer low side is connected to a bus that is supplying radial feeders, the de-energization of the transformer will cause the de-energization of that bus and the tripping of all feeder breakers.

Case c: Line out of service

The following alarms will be observed when a line between substations 1 and 2 goes out of service:

- Line MW reading drop at substation 1 < E1.
- Line MW reading drop at substation 2 < E1.
- Line MVAR reading drop at substation 1 < E2.
- Line MVAR reading drop at substation 2 < E2.
- Line disconnected (DC) at substation (EMS Message).
- Line disconnected (DC) at substation 2 (EMS Message).

- Line de-energized (DE) (EMS Message)
- Breaker (or Breakers) trip at S/S 1.
- Breaker (or Breakers) trip at S/S 2.
- Voltage value drop to < E3 at S/S 1.
- Voltage value drop to < E3 at S/S 2.

Where E1, E2, and E3 are as defined above. The "EMS message" refers to the type of alarms that are generated by the EMS programs and not telemetered from the field.

Case d: <u>Line open at one end</u>

Following alarms are observed when a line is disconnected at one end:

- Line MW value drop to < E1.
- Line MVAR value drop to < E2.
- Line Voltage value drop to < E3.
- Breaker (or Breakers) trip.
- Line disconnected (DC) (EMS message).

Where E1, E2, and E3 are as defined above.

Case e: <u>**Bus de-energized**</u>

Following alarms are observed when a bus is de-energized:

- Tripping of all breakers connected to the bus.
- Bus Voltage value drop to < E3.
- Bus Frequency value drop to < E4.

Where E3 and E4 are as defined above.

The de-energization of a bus is also accompanied by other alarms, occasionally. The type of these alarms vary depending on the bus loading at the disturbance time, bus type (generator or load bus) and other factors. This issue will be discussed in later sections.

Case f: <u>**Loss of load**</u>

This disturbance is usually observed under different scenarios and more than one set of alarms can represent the event, however, we have considered the following general case:-

- Load line breakers tripping.
- MW value drop.
- Voltage alarms.
- Frequency alarm (if load is significant).

Case g: Voltage high or low at a substation

This situation is realized when voltage varies due to significant load, generation, or VAR variations. Under such circumstances the dispatcher will observe many alarms (one at each telemetered point, i.e line,bus, transformer,etc..). All alarms will contain the same information but they will be from different field points (i.e. power systems equipment in the substation). Therefore, the alarms received for this case are one voltage alarm per each telemetered equipment in any station.

3.2.2 Non-Consistent Alarms

There are other alarms that usually accompany the indicative alarms but they are not considered decisive for the occurrence of a particular disturbance on the network. For instance, when a bus is de-energized the indicative alarms will be accompanied by some other alarms. For example, SF6 urgent alarm, or MW, MVAR and KV alarms in nearby buses and substations. These alarms are not considered indicative alarms because they may be generated in some instances but will not appear in others. For this reason they are referred to here as non-consistent alarms. These alarms can be divided into two categories:-

1. Alarms that the dispatcher does not need to see (related alarms). These are the alarms that pertain to equipment which have failed or was a cause of the

disturbance. This type of alarms can be useful after the event and should not affect the analysis of what is happening on the network during a major power system disturbance. They are stored in a historical file and suppressed from display. Typical alarms of this type would be:

- Lines: Loss of AC/DC to different Relay Sets.
- Buses, Breakers and Transformers: Oil pressure alarms, SF6 urgent and non-urgent alarms, relay operation alarms, etc...

One important reason for suppressing these alarms from display during power system disturbance is that the dispatcher cares more about what have failed, to what extent the network was affected, and how to relieve the system. Therefore, only information related to these topics need to be displayed when a disturbance occurs. The rest of the alarms information is useful to diagnose the failed equipment and can be referred to in a historical file any time.

It is important to know that the suppression of these alarms should be done only when these alarms accompany indicative alarms during power system disturbances on the network. However, when any of these alarms is generated on individual basis or without any relation to a particular disturbance it should be displayed as is. The IAP will only suppress this type of alarms on a particular equipment when a conclusion can be derived from the alarm data on that equipment.

2. Alarms that the dispatcher needs to see (unrelated alarms).

   Unlike previous type of alarms, these are the ones that pertain to the network configuration and status. The dispatcher needs to see them even during a power system disturbance. The information supplied by these alarms are needed, as much as the conclusions derived by the IAP, to restore the network to its normal state. As a result, these alarms need to be displayed together with any conclusion that may be reached by the IAP. Typically, these alarms are MW, frequency, MVAR and Voltage changes due to loss of lines, buses and generators. However, voltage alarms will still be governed by empirical rules for high and low voltage, i.e. in cases where a major load or a generator is dropped from the network, effect of voltage may spread to several substations and many alarms will be produced at each substation - one for each telemetered equipment. Despite that the IAP will produce a message about the major load or the generator drop, the dispatcher will also need to know the affected substations. However, the dispatcher does not need to see an alarm for each particular equipment, where they can be summarized in one message for the whole substation.

3.3 Power System Connectivity Data Base

Upon receiving alarms at the PCC, dispatchers first perform an analysis of received alarms and try to relate them to each other. This analysis is made in an effort to pinpoint a particular disturbance or disturbances that the system may have undertaken at that time. While analyzing the incoming alarms the dispatchers also observe the system diagram, by looking at another screen or on the map-board. This helps them to draw relationships between equipments under alarm and their connectivity to each other and to the rest of the system as well. For instance, if three alarms were received at a substation for breaker 5, Line 1 and for Bus 2. The dispatcher needs to visualize how the breaker, the line and the bus are connected to each other before making a conclusion. Consequently, the substation connectivity data tables constitute an important integral part of any expert system . Such tables are usually available in the data base software at modern power control centers. The connectivity tables in the PCC data base describe the network connectivity in terms of bus-sections and circuit breakers. All equipment, such as generators, transformers, transmission lines, etc., are connected to bus-sections. Bus-sections within one-voltage level at a substation may be connected together by a breaker [25].

The connectivity tables contain two sets of information. One set for the assignment of unique connectivity numbers for each bus-section. The other set includes the designation of each breaker as 'from' and 'to' which bus-section it is connected. Tables 1 and 2 display the bus-section numbers and the breaker connectivity data for the diagram shown in figure 5.

TABLE 1. Bus-section Connectivity data for the sample substation shown in figure (6)

S/S	Connectivity No	Equip. Type	Identification
1	1	Bus	A
1	2	Bus	B
1	3	Line	LT1
1	4	Transformer	TR1
1	5	Transformer	TR1
1	6	Bus	C
1	7	Generator	GU1

TABLE 2. Breakers Connectivity data for

the sample substation shown in figure (6)

S/S	BRKR No.	From	To	Status
1	01	1	3	Close
1	02	2	3	Open
1	03	1	4	Close
1	04	2	4	Close
1	05	1	6	Close
1	06	6	7	Close

S/S = Substation

BRKR = Breaker

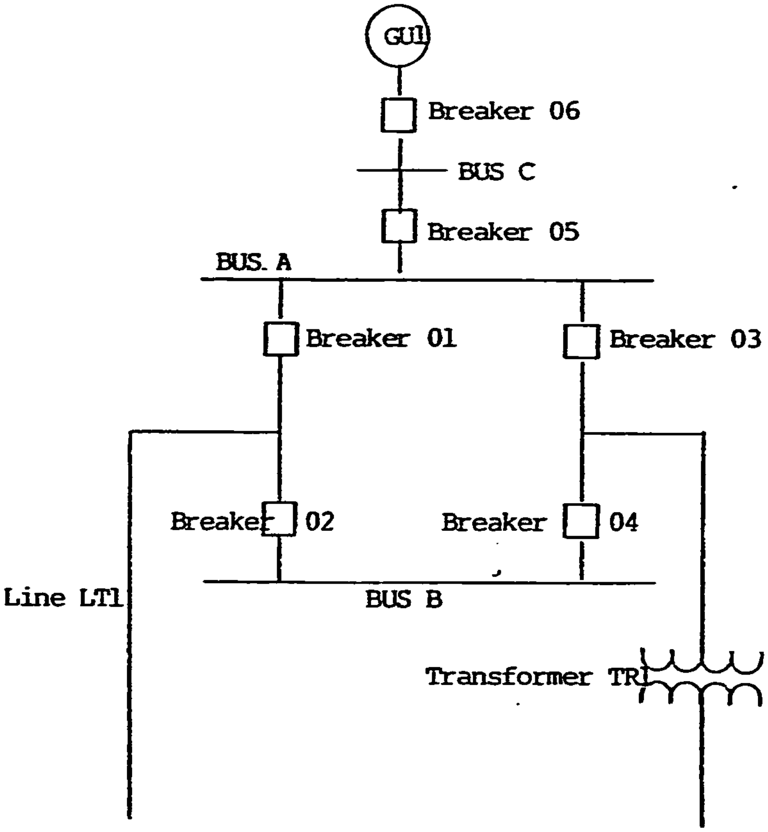


Figure 5 : A sample substation diagram

To emulate a dispatcher at a PCC, the IAP needs the connectivity relation between various equipment in the substations. Therefore, the IAP should access substations connectivity tables or interface with these data file all the time.

3.4 BUILDING A KNOWLEDGE RULE-BASE PROTOTYPE

The group of alarms shown in section 3.2 represent sets of indicative alarms. When a set is observed by a dispatcher at the PCC, a disturbance case on the network is identified. The key tasks of the IAP, in sequence, are:

- Realization of a possible power system disturbance through scanning the input alarms.
- Confirming the existence of a problem or problems once realized.
- Extraction of all alarms related to the disturbance.
- Displaying a conclusion regarding the derived problems.
- Suppressing alarms, that the dispatcher does not need to see, and displaying other alarms that either no conclusions were deduced form or the dispatcher needs to see, together with any reached conclusion.
- Storing all suppressed alarms in a historical file.

This section reviews briefly each task. In addition the rules that are built to accomplish each task for every major disturbances, mentioned in section 3.2, will be provided.

3.4.1 Realization Of A Possible Disturbance

In real time operations, such as in PCC, time is very important. It will be time consuming to match every set of alarms received at the PCC for a possible match with all pre-defined alarm groups. Therefore, criteria were set for each group of alarms. If these criteria were satisfied the particular disturbance represented by this group of alarms can be assumed. Further confirmation will be made afterwards to validate this assumption.

Upon examining the indicative alarms defined for each major disturbance in section 3.2, one can divide the alarms for each case into two types. These two types are the analog value alarms and the breaker tripping alarms. This characteristic applies to all cases except the high and low voltage cases. Such a breakdown was utilized to define the realization and confirmation criteria. The criterion used to realize the occurrence of a possible disturbance utilizes the alarms generated for the analog values on each equipment. On the other hand, confirmation is achieved through verifying that breaker tripping alarms were generated for all breakers connected to the equipment under investigation. The reason for selecting the alarms on analog values as the criteria for the disturbance, as opposed to breaker's alarms, is the fact that verification of breakers alarms requires more time. This is due to the fact that an extraction of all connectivity information, which consumes relatively longer time, will have to be performed first. Thus, it does not need to be done at elementary stages when IAP is still looking for possible disturbances on the network.

Example rules for the generator off-line case will be as follows: For the generator shown in figure 6 there will be 6 indicative alarms when it goes off-line. Four analog value alarms and two breakers tripping alarms. The realization rules for this case will be:

```
(IF     Alarm 1  at S/S 1 on Gen A for MW   value = X
        AND (X < E1)
        AND  Alarm 2  at S/S 1 on Gen A for MVAR value = Y
        AND (Y < E2)
        AND Alarm 3  at S/S 1 on Gen A for KV   value = Z
        AND (Z < E3)
        AND Alarm 4  at S/S 1 on Gen A for HZ   value = W
        AND (W < E4)
THEN
        Gen A at S/S 1 is assumed off-line)                    (Rule 3.1)

(IF     Gen A at S/S 1 is assumed off-line
        AND Alarm at S/S 1 for all BRKR's (iA) TRIP
THEN          Gen A at S/S 1 is OFF-LINE)                      (Rule 3.2)
```

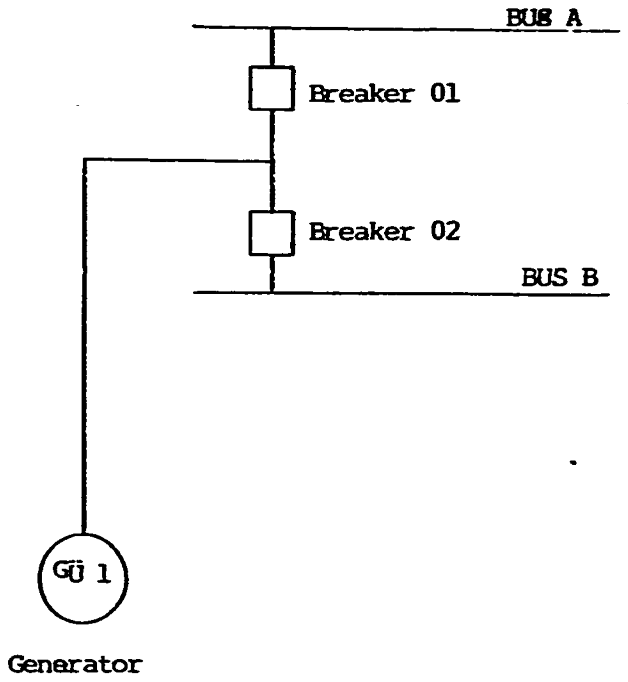


Figure 6: A generator connectivity diagram

Where X,Y,Z, and W are the actual MW, MVAR, Voltage, and Frequency readings at the time the alarms were generated. E1, E2, E3, and E4 are error value readings assumed in the telemetry devices of the MW, MVAR, KV, and Frequency, respectively. HZ stands for frequency, Gen for generator, KV for voltage, and S/S for substation.

Note that the empirical rule includes two subrules. The first rule (3.1) is responsible for the realization of a possible specific disturbance. The IF part of this subrule, the antecedents, includes alarms on analog value readings, where the THEN part, the consequent, is responsible for generating the fact designating the possible occurrence of a disturbance on a particular equipment. The second rule (3.2) is the confirmation part and will be discussed in detail in the next section.

The realization rule can be stated in literature for the example under consideration as follows. The generator A at substation 1 is assumed off-line when the four indicative alarms on analog values are available in the tested input alarms. In addition, each of the MW, MVAR, KV, and frequency readings, at the time alarms were generated, is approximately zero. Since there are usually error readings in the telemetry devices, low error values were allowed for the inaccuracy in these devices.

3.4.2 Confirmation Of A Power System Disturbance

As it was stated above after realizing that there is a possible disturbance, a confirmation or assumption validation cycle follows. In this cycle, all breaker's tripping alarms found in the particular substation under investigation, are extracted. Then, a cross reference check is performed with the connectivity data base tables to verify that all breakers connected to the particular equipment under investigation have tripped.

The reasoning process in this cycle is described as follows:

1) The realization rule asserts, adds, new fact to the fact-list, agenda, once an assumption has been made. This new fact pertains information about the disturbance type, substation name, equipment type, and identification.
2) The new fact will match the first pattern of the confirmation rule for a specific disturbance forcing an engagement of that rule (each statement of the IF antecedents is called pattern in CLIPS) . Then, the confirmation rule utilizes the information provided by this fact, namely the substation name, equipment type, and identification, to acquire the equipment unique connectivity number 'n' from the connectivity data base.
3) This will be followed by an extraction of all breakers connected to the equipment in the troubled substation. This information will also be extracted

from the data base connectivity tables . The recognition of breakers that are connected to bus-section 'n' is achieved by finding all breakers in that particular substation such that the number 'n' matches either the 'TO' or ' FROM ' number of that breaker. All breakers are designated by two connectivity numbers as explained earlier.

4) Finally the assumption is validated once it is verified that alarms were generated for each of the breakers identified. This concludes the confirmation rule.

The reasoning process discussed above can be further illustrated by looking at the connectivity numbers in figure 7. If bus A is under disturbance confirmation cycle, the number 'n ' will first be realized. Breakers m and k have their connectivity numbers as 'n', 'n+1' and 'n', 'n+2', respectively. Since each of these two breakers have the number 'n' in its connectivity representation, they will be extracted and the program will return with there identifications, i.e m and k. Following is a general rule structure defining the confirmation logic:

Rule 1:

IF - Flag A was set (indicating that an assumption was made on probable occurrence of disturbance X on equipment Y at S/S J) (pattern 1)

- extract connectivity number for Y (n) (pattern 2)

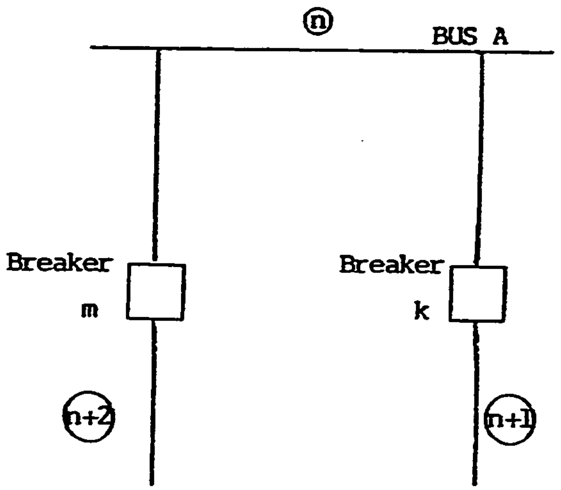


Figure 7: Example for set of breakers in Confirmation process

* The ringed numbers refere to the connectivity assignment of the relative bus section.

```
  -  extract all breaker's [z] identification which are connected 'TO' or 'FROM'
     bus-section 'n' in S/S J                                        (pattern 3)
  -  Verify that all breakers [z] have alarms generated              (pattern 4)
THEN
  -  Assumption (disturbance x) is confirmed.                         (action 1)
ELSE
  -  Assumption is not true.                                          (action 2)
AND   Display all alarms.                                (action 3)  (Rule 3.3)
```

Figure 8 depicts the realization and confirmation logic.

3.4.3 Conclusion Display And Alarm Suppression

The IAP is designed to display the conclusions or intelligent messages that are determined immediately once they are validated. The message will contain the time, date, substation name, and information determined about the disturbance or abnormality in a certain equipment. The indicative alarms used to determine the reached conclusion will not be displayed. However, it will be useful to keep all received alarms in a historical file, for future reference. A software file was introduced that will keep track of all received alarms. This file simulates a logger

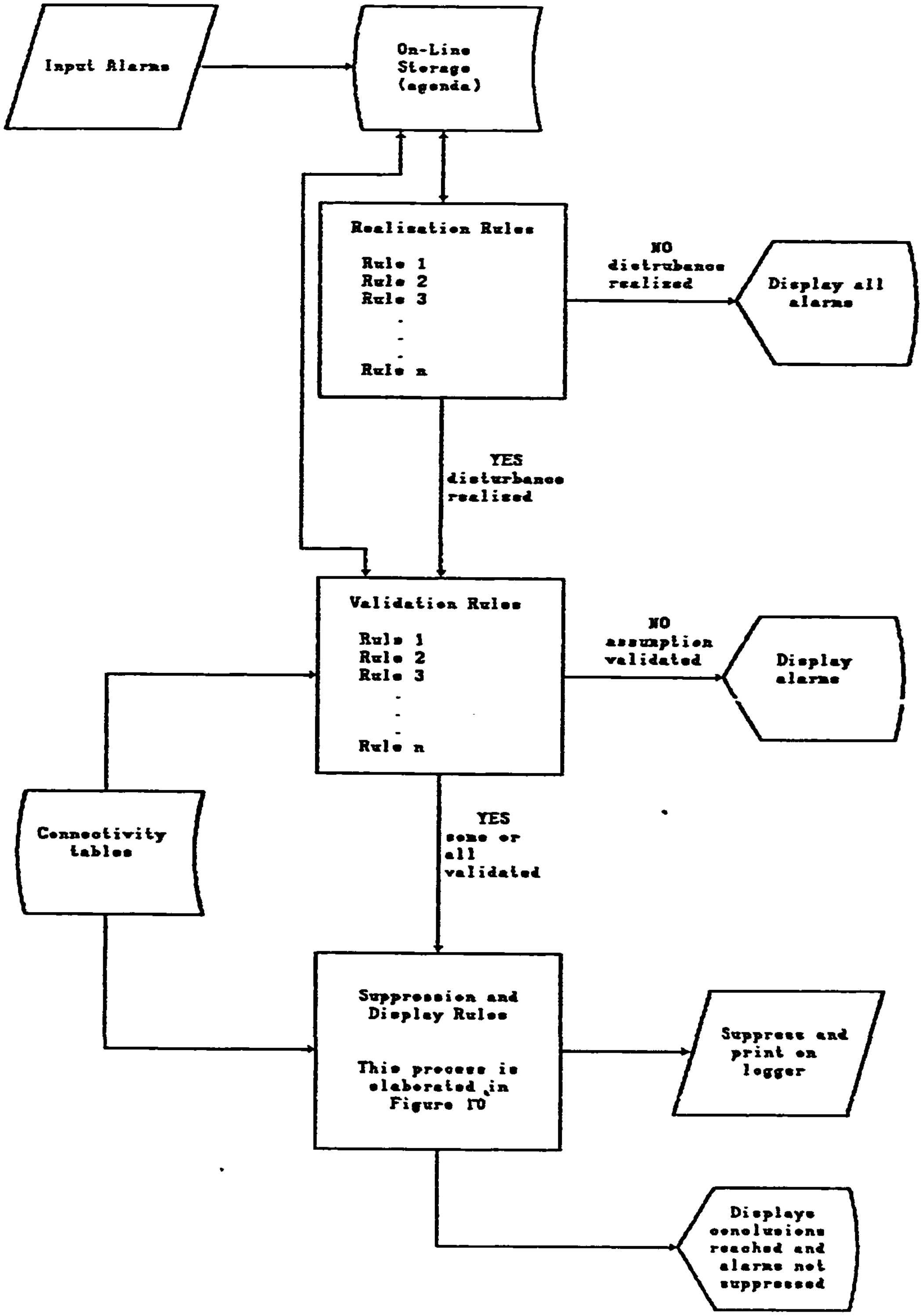


Figure 8: The Realization and Confirmation Process

(printer) in Power Control Centers where all alarms will be printed as received from the field.

The criterion for alarms suppression is that any redundant information that can by assumed when stating that a certain equipment fails or experiences a particular disturbance should not be displayed. This will include all indicative alarms that were used to determine the disturbance. In addition, all other alarms that may be generated on that equipment are not to be displayed, this includes the non-consistent related alarms as defined in section 3.2.2.

The non-consistent alarms of the second category, unrelated alarms, defined in section 3.2.2, and other alarms that no conclusion could have been deduced from will all be displayed together with the conclusions that.were reached, if any.

Redundant and non-consistent related alarms are suppressed as soon as a conclusion is reached on a particular power system equipment except the breakers tripping alarms. These are kept until all confirmation rules are processed. The main reason behind this strategy is that breakers are common between various power system equipments in substations. While one conclusion may have been reached for an equipment, a contiguous equipment, that has one breaker connected to both, may also be tested for a probable disturbance. In this case, the non-existence of alarm on that breaker will not validate the probable conclusion.

The rule for this process is as follows:

IF Disturbance X occurred at Equip. A in S/S. B – pattern 1
AND Alarm z at S/S B Equip. A ?$ Y (any text) – pattern 2
THEN Delete Alarm z (Rule 3.4)

Note that this is a recursive type rule which will keep firing (running) until there are no more alarms that matches the second pattern, i.e alarm z.

This rule can also be stated as follows:

IF there is a fact in the fact-list that confirms a disturbance 'X' in substation 'B' at equipment 'A'.
AND there exists an alarm on the same equipment at the same substation that carry any message ?$ Y, i.e. MW, or KV or SF 6 urgent, etc...THEN delete the alarm found. The ?$ Y is a variable type assignment in CLIPS that can match with any text that consists of multiple words in a conditional fact.

Once the alarm is deleted the rule will backtrack trying to satisfy the antecedents with different fact, should there be any. The antecedents will be satisfied if there is another alarm in the fact-list which meets the conditions stated above. Therefore, the rule will keep iterating until all alarms related to that particular equipment are deleted.

Note that while the second pattern will be satisfied for all indicative alarms on any disturbed equipment, it will also be satisfied for non-consistent related alarms on that equipment. Figure 9 depicts the display and suppression process in the IAP prototype.

3.4.4 Constraints, Assumptions And Limits

Electric power utilities have different standards for telemetry points readings. While some utilities specify analog and status telemetry on each power system equipment in the substation, other will select some only. The following specifications are assumed throughout the thesis:

1) All buses voltage and frequency values are telemetered.
2) The Mega Watts (MW), Mega Vars (MVAR) and the voltage (KV) values of all lines and transformers are monitored.
3) The MW, MVAR, KV, and frequency (HZ) values of all generators are telemetered.
4) The status of all breakers are monitored.
5) Both high and low sides of the transformers KV values are telemetered. However, telemetry devices for MW and MVAR are installed at one side only of the transformers, namely the high side.

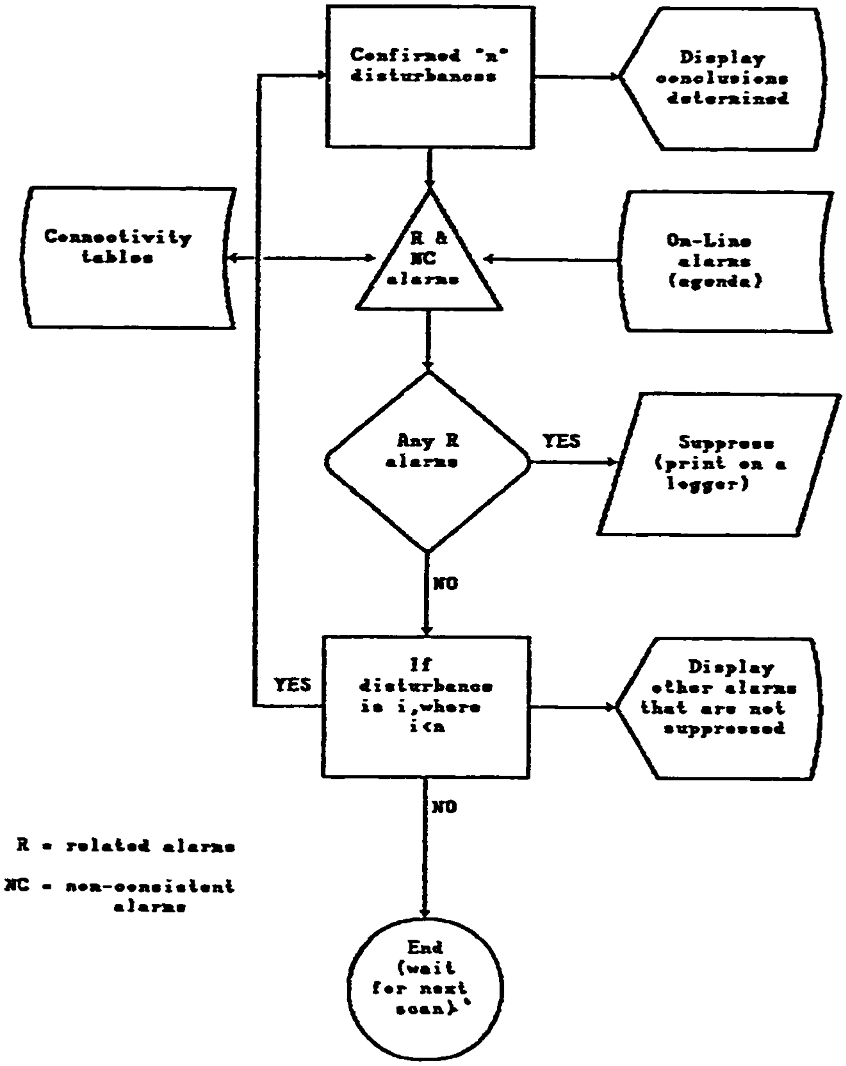


Figure 9 : The Suppression and Display Process

6) The alarm processor at the PCC will generate a message or an alarm for any status or reasonable analog change of all telemetered points.

The following assumptions were also made in building the knowledge-base.

1) Breakers status change alarms are considered more reliable than analog values, therefore no effort was made to build rules to detect their malfunction operation. This assumption has been made for the following considerations:-

   a) Analog metering devices are the ones that fails more frequent and need calibration more often.
   b) Most PCC's have a way to verify that breakers control is not functioning by receiving an alarm whenever a control command is initiated and no response occurs (command-no-take case). Therefore, there is another reliable way of verifying the reliability of the breaker's telemetry.
   c) The visual verification of breakers controls instrumentation failure is much easier than analog, because it does not invoke accuracy. It is either ON or OFF which can also be verified by dispatchers most of the time.

2) The accuracy of all analog metering devices was assumed 1%. This can be set in the IAP to any desired value.

3) The data base should possess connectivity information. This is generally a valid assumption since it is needed for the real-time modeling of the power system.

4) The IAP is designed to process the data only if the number of the power system alarms received at the PCC during a full scan time exceeds 3 alarms. This limit has been set since human intelligence can analyze the small number of alarms easily without a need for an aid.

5) Alarms generated during a full scan period will be queued in a file, namely MSG. DAT, and at the end of each scan period, the scanning protocol will issue a command that will initiate the IAP program. The IAP will then display the results, suppress some alarms, and print all received alarms on the intelligent basis that it was designed for.

6) Only major disturbances that could cause a huge number of alarms, as defined by the questioned dispatchers, were considered.

3.4.5 Empirical Rules

These rules designate the core rules by which the IAP will run and on which the knowledge base will be built on to emulate power system dispatcher's way of analyzing alarms in reaching a conclusion. In this section rules of each of the nine disturbance cases, defined earlier, will be presented. Rules will be represented in the IF-THEN logic, i.e. IF set of alarms exist, THEN case A is true. Further, the rules for each case are divided into two rules. the first being the realization rule and the second is the confirmation rule. The correlation of all rules to each other and the formation of the final knowledge base will be explained in the following sections.

CASE 1: Generator Off-Line (OL)

Following are rules that govern the detection of a generator off-line
(Date, time, and the word 'alarm ' are omitted from the alarms representation in the IF part of the rule for simplicity):

```
(IF     S/S A  Gen B   MW = X
   AND  X < E1
   AND  S/S A  Gen B MVAR = Y
   AND  Y < E2
   AND  S/S A  Gen B   HZ = Z
```

AND Z < E3

AND S/S A Gen B KV = W

AND W < E4

THEN

S/S A Gen B is assumed off-line) (Rule 3.5)

(IF S/S A Gen B is assumed off-line

AND S/S A BRKRS(ib) TRIP

THEN

Gen B at S/S A is OFF-Line) (Rule 3.6)

Where X,Y,Z & W are actual MW, MVAR, Frequency and voltage readings at the time the alarms were triggered. E1, E2, E3 and E4 error value readings assumed in the telemetry devices of MW, MVAR, Frequency, and Voltage values, respectively. Gen is a symbol and refers to generator. E(i) is calculated as per the following equation:

$$Ei = a(v) * N(i)$$

Where a(v) stands for percentage accuracy of the telemetry device 'v' and N(i) for nominal values at equipment 'i' (for frequency and voltage readings) and full rated value (of MW and MVA readings).

The rules presented above were explained in section 3.3.1 and were repeated here for the purpose of completing all the cases.

CASE 2: Bus De-Energized (DE)

In spite that the de-energization of a bus will, usually cause wide spread of alarms among the different power system components, there are specific alarms that shall appear all the time this disturbance occurs. Consequently, these specific alarms are considered the defining group or indicative alarms for this case. These are the bus frequency drop alarm, the bus voltage drop alarm, and the tripping of all breakers connected to the bus.

The empirical rules for this case are defined as follows:-

```
(IF          S/S A  Bus B   HZ = Z
       AND   Z < E1
       AND   S/S A  Bus B   KV = W
       AND   W < E2
THEN
             S/S A  Bus B is assumed De-energized (DE) )          (Rule 3.7)

(IF          S/S A  Bus B is assumed DE
       AND   S/S A  BRKRS(ib)   TRIP
```

THEN

Bus B at S/S A is DE) (Rule 3.8)

Where Z & W are actual frequency and voltage readings at the time alarms were depicted. BRKRS(ib) refers to all breakers connected to bus B. E1, E2 are error readings assumed in the telemetry devices of Frequency, and Voltage values, respectively.

The rules can be restated as follows.

Rule 3.7: If alarm 1 is triggered on bus B at substation A for frequency drop, value is Z.

And alarm 2 is triggered on bus B at substation A for voltage drop, value is W.

Where Z is less than E1 And W is less than E2.

Then Assume bus B at substation A is De-energized (assert a new fact to the fact-list)

Rule 3.8: IF Bus B at substation A is assumed De-energized

And Alarms are triggered for all breakers connected to bus B at substation A

Then Bus B at substation A is De-energized (Display)

CASE 3: <u>Line Open Ended (OE)</u>

This case occurs when a line is disconnected at one substation. The set of indicative alarms that are expected to be triggered in this case were provided in section 3.2.1 case d. Following are the empirical rules representing the realization and confirmation of this case:-

```
(IF         S/S A  Line B   MW = X
      AND   X < E1
      AND   S/S A  Line B MVAR = Y
      AND   Y < E2
      AND   S/S A  Line B   KV = W
      AND   W < E3
      AND   S/S A  Line B  DC (EMS)
THEN
            S/S A Line B is assumed OE)                    (Rule 3.9)
```

```
(IF         S/S A Line B is assumed OE
      AND   S/S A  BRKRS(ib)  TRIP
THEN
            Line B at S/S A is OE)                         (Rule 3.10)
```

Where X,Y, and W are actual MW, MVAR, and voltage readings, respectively at the time alarms were depicted.

The last pattern of rule 3.9 is not a telemetered message,i.e. received from the substation. It is a message generated by the EMS software , particularly by the network status processor (NSP) program, whenever a line is disconnected.

CASE 4: <u>Line Out Of Service (OOS)</u>

This case is similar to the open ended case except that another set of alarms are generated at the other end of the line, i.e. the other substation that the line is connected to. The empirical rules for this case are as follows:-

```
(IF          S/S A1  Line B   MW = X1
       AND   X1 < E1
       AND   S/S A1  Line B MVAR = Y1
       AND   Y1 < E2
       AND   S/S A1  Line B   KV = W1
       AND   W1 < E3
       AND   S/S A2  Line B   MW = X2
       AND   X2 < E1
       AND   S/S A1  Line B MVAR = Y2
       AND   Y2 < E2
       AND   S/S A1  Line B   KV = W2
```

AND W2 < E3

AND S/S A1 Line B DC

AND S/S A2 Line B DC

AND Line B DE

THEN

Line B is assumed (OOS) between S/S's A1 and A2) (Rule 3.11)

(IF Line B is assumed OOS between S/S's A1 and

AND S/S A1 BRKRS(ib)1 TRIP

AND S/S A2 BRKRS(ib)2 TRIP

THEN

Line B between S/S's A1 and A2 is OOS) (Rule 3.12)

The last three patterns in rule 3.11 are EMS messages that are produced by NSP program whenever a line is out of service.

CASE 5: Transformer De-Energized (DE)

The transformer is defined as de-energized when the primary side - High Side (HS) - and the secondary side - Low Side (LS) - are both disconnected from other components in the substation. The following empirical rules describe this disturbance realization and confirmation:

```
(IF         S/S A  Trans B   MW (HS) = X1
      AND   X1 < E1
      AND   S/S A  Trans B MVAR (HS) = Y1
      AND   Y1 < E2
      AND   S/S A  Trans B   KV (HS) = W1
      AND   W1 < E3
      AND   S/S A  Trans B   KV (LS) = W2
      AND   W2 < E4
THEN
            Trans B is assumed (DE) at S/S A)                    (Rule 3.13)
```

```
(IF         Trans B is assumed DE at S/S  A
      AND   S/S A  BRKRS(ib) HS   TRIP
      AND   S/S A  BRKRS(ib) LS   TRIP
THEN
            Trans B at S/S A is DE)                              (Rule 3.14)
```

CASE 6: High Voltage At A Substation (HV)

Since under this case the dispatcher will observe many alarms (one for each telemetered equipment at the substation, i.e line, bus, transformer, etc..). All alarms will contain the same information but they will be from different field points . What the dispatcher needs to know is that substation X is under high or low voltage and value is n.

A criterion was defined to detect this case as follows. If there exist alarms on at least two different buses in a single substation such that the actual voltage value of each of them is greater than the nominal voltage value, and the differences between the readings do not exceed 1%, then, the high voltage disturbance case is assumed. The empirical rules governing this case would be as follows:

```
(IF         S/S A Bus B  KV(Nom.) Voltage = n            -- pattern 1
      AND   n > KV(nom.)                                 -- pattern 2
      AND   S/S A Bus D  KV(Nom.) Voltage = m            -- pattern 3
      AND   n > KV(nom.)                                 -- pattern 4
      AND   D NEQ B                                      -- pattern 5
      AND   m < n + (Nom * 1%)                           -- pattern 6
      Or    m > n - (Nom * 1%)                           -- pattern 7
Then
            S/S A is under HV = n)                       (Rule 3.15)
```

Where each independent statement in CLIPS is called pattern.

The rules presented above can be explained as follows.

- pattern 1: alarm at S/S A on bus B of nominal voltage = KV(nom.) is triggered for voltage reading = n.
- pattern 2: n is greater than KV(nom.)
- Pattern 3: alarm at S/S A on bus D of nominal voltage = KV(nom.) is triggered for voltage reading = n.
- Pattern 4: n is greater than KV(nom.)
- Pattern 5: Bus B is not Bus D.
- Pattern 6 & 7: These patterns Check that The difference between the two readings is within 1% range. This is the minimum inaccuracy assumed in the telemetry devices.

CASE 7: Low Voltage At A Substation (LV)

The discussion for the high voltage case holds true for the low voltage case, except for replacing the word low for high and reversing the comparison signs '>' and '<' where ever they appear in the text.

CASE 8: Loss of Load (LOL)

The more frequent loss of load case that was selected in section 3.2.2 for the IAP implementation have the following as indicative alarms:

- Alarm on breaker tripping of the load line.
- Alarm on MW value drop for the load line.
- Alarm on KV drop for the load line
- Alarm, or alarms , on KV rise in the substation (If lost load is significant).

The conditional presence of the last two alarms requires defining two separate realization rules for this case. One if the load is significant and the other if it is not. These two rules will be introduced first by text, for clarity, and then the symbolic empirical rules will be presented.

1)If alarm 1 is triggered at S/S A on load B for MW drop to x
and alarm 2 is triggered at S/S A on load B for MVAR drop to y
and alarm 3 is triggered at S/S A on load B.for KV drop to w
where x, y, and z are less than E1, E2, and E3, respectively.
Then assume loss of load B at S/S A.

2)If loss of load B at S/S A is not assumed
and alarm 1 and alarm 2 and alarm 3 are triggered at S/S A
and (any alarm is triggered for high voltage on any equipment, at S/S A
or high voltage is defined at S/S A (as determined by case 6)).
Then assume loss of load B at S/S A.

3)If loss of load B is assumed at S/S A

and All breakers connected to load B tripped

Then Load B at S/S A is lost

Note that the first check in the second rule verifies that the first rule did not fire. Should the first rule succeeds to make an assumption on load B, the second rule will not be encountered since first pattern of the second rule will fail. The empirical rules in symbols for this case are as follows:

(IF S/S A Load B MW = X

AND X < E1

AND S/S A Load B MVAR = Y

AND Y < E2

AND S/S A Load B KV = W

AND W < E3

THEN

Loss of Load B is assumed at S/S A) (Rule 3.16)

(IF NOT Loss of load B is assumed at S/S A

S/S A Load B MW = X

AND X < E1

AND S/S A Load B MVAR = Y

AND Y < E2

```
     AND  S/S A  Load B   KV = W

     AND  W < E3

     AND  (S/S A  ?equip  KV  = Z

          Z > KV(nom.)

     OR   S/S A in under High Voltage value = Z)

THEN

     Loss of Load B is assumed at S/S A)                    (Rule 3.17)


(IF  Loss of Load B is assumed at S/S A

AND  S/S A  BRKRS(ib)   TRIP

THEN

     Loss of Load B at S/S A)                               (Rule 3.18)
```

CASE 9: <u>Partial System Black Out (SBO)</u>

The rules for this case are different from the rules for the previous cases. They concern several equipment at, relatively, one time. The IAP recognizes one or more substations that experience black out and it will list all these stations. For a substation to undergo or experience a black out, all buses must be de-energized. However, some of these buses may already be on a scheduled outage and no alarms will be produced for these buses when a black out occurs. An initial requirement was set to start realization process of the occurrence of a black out. This requirement is the realization of a minimum of one bus de-energized and a line out of service or open ended at that substation. Once the IAP realizes that this requirement has been met for a substation, it will set a flag in the agenda (by introducing a new fact). This flag, in return, will match the first antecedent or pre-condition of the IF part of the first rule for the black out forcing the engagement of that rule. Realization of bus de-energized and line out of service was explained earlier in cases 2 and 4, respectively. Once the first rule of the black out is engaged, IAP checks on all other lines if they are either out of service, open ended, or on a previous scheduled outage where no alarm could exist in the last case.

IAP decides that a line is on a planned outage when all breakers connected to that bus are open, such information are verified through the connectivity data-base tables. The set of rules, describing this sequence is provided next:

Rule 1: IF IAP detects a Bus de-energized and a line out of service at S/S J

THEN Add a new fact (setting a Flag 1) for a possible black out in S/S J

(Rule 3.19)

Rule 2: IF Flag 1 is set on the agenda

THEN extract names and information on all lines connected to S/S J , define such information in $x. (Rule 3.20)

(Where $X designates a vector variable which contains multiple variables in Clips).

Rule 3: IF All lines $X at substation J

and Lines are out-of-service or open ended or disconnected at either or both ends of each line.

THEN substation J is islanded or blacked out. (Rule 3.21)

Rule 3 will verify that all incoming and outgoing lines are either out-of-service, open ended or disconnected earlier, i.e. no alarm was produced at that time due to earlier line trip or scheduled outage on that line.

The verification of earlier disconnection of a line and the extraction of identification of all lines in the substation are performed through connectivity database which possess the up-to-date breaker's status.

3.5 GENERALIZATION OF THE PROTOTYPE (IAP)

In spite that the components of the IAP established so far handles the power system disturbances realization and confirmation from the input alarms to the PCC, it still requires additional knowledge to make it more flexible and to accommodate special cases. Some rules have to be established to initiate the operation of the IAP. Another issue to consider is when a power system disturbance occurs and not all alarms expected to appear were generated. This could be due to bad telemetry devices or bad wiring of sensory points at the substation. All these are considered in this section.

3.5.1 Incomplete Group Of Alarms

Alarms are reported to the PCC from various power system substations by the Remote Terminal Unit (RTU). RTU, in its turn, collects information about all substation equipment through telemetry devices or sensory circuits. If any of these telemetry devices fails, it will either report wrong information or will not report any information. However, in either case, if the dispatcher failed to recognize the failure of any of the telemetry devices an alarm may be received where it should not or no alarm will be received where it should. This could be confusing to the dispatcher and the IAP because the group of indicative alarms may not be all generated when a disturbance occurs. Bad telemetry devices can only be recognized when a disturbance occurs and a set of alarms is expected to be reported to the PCC but not all alarms are reported or when some are reported but incorrectly. A solution to this is to establish rules that can realize the possible occurrence of a disturbance from subsets of the group of indicative alarms. Confirmation through breaker tripping alarms can validate the assumption of the disturbance occurrence. Further, the rules will indicate the existence of bad telemetry devices on the disturbed equipment.

The empirical rules governing this reasoning process, in general, will be as follows:-

Rule 1 IF NOTdisturbance X is determined on equip B at s/s J

AND (alarm A1 on equip B at s/s J

OR alarm A2 on equip B at s/s J

OR ________________)

THEN Disturbance X is assumed at s/s J with bad telemetry device (BTD) (Rule 3.22)

Rule 2 IF Disturbance X is assumed on equip B with BTD at s/s J

AND BRKRiB, all connecting breakers, trip at s/s J

THEN display "disturbance X on equip B at s/s J with BTD" (Rule 3.23)

The first pattern of Rule 1 is intended to check that no conclusion was made earlier for the same case but with a complete alarm group. Since the "or" logic will also hold true even if all alarms exist,i.e when the group alarms are complete, it is necessary to rule out the case when a conclusion was already reached for any set of alarms. This general logic for handling incomplete group of alarms is depicted in Figure 10.

The empirical rules described above apply to all of the cases except to the high and low voltage cases. This is due to the fact that the number of indicative alarms for each of these two cases is not fixed. For example, the number of indicative alarms that are generated for a substation consisting of 2 lines, 2 buses, and one transformer is 6 alarms. However, this number will be different for another

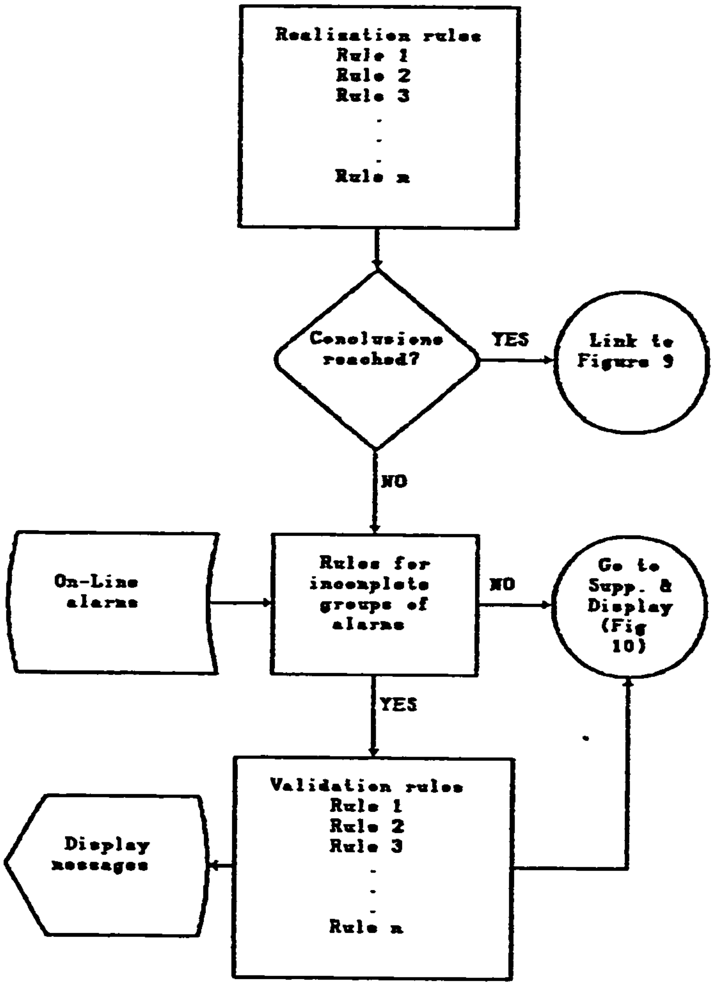


Figure 10: Handling Incomplete Groups of Alarms

substation that consists of 4 lines, 2 transformers, and 2 buses, namely 10 alarms. The empirical rules for these two cases is as follows.

IF KV alarm on equip x at S/S A value = KV1
AND Equip ?x1 (NEQ x) at S/S A
(NOT) KV alarm on ?x1 at S/S A
THEN HV or LV at S/S A with some BTD (Rule 3.24)

Rule 3.24 can be restated as follows:

IF there exist a voltage alarm at equipment x at substation A
AND there is an equipment ?x1 other than x (the '?' sign designates a variable that can apply to any equipment) in S/S A
AND no alarm is generated for the equipment ?x1
THEN Voltage (high or low) at S/S A with some non-operational telemetry devices.

3.5.2 The Initialization Of The IAP

The IAP is initialized by a batch program which was developed to serve as the interface program between the PCC software and the IAP. Upon the completion of a full scan for all substation's RTU's, which typically occurs every 2 seconds, the IAP is initialized and the initiation program will run. This can be implemented in real-time situation by modifying the scan protocol of the Data Acquisition subsystem at the PCC. The modified protocol will send a command that will start the initiation program (called here init. bat) at the end of each scan. Once init. bat is initiated, it performs the following functions in a sequential manner:-

1) Read alarms, from the alarm list file (called here msg. dat).

2) Count the number of alarms. If number of alarms is less than 3, the initiation program will display the alarms without being passed to the IAP. This criterion was established to assure that the IAP will not run if the number of alarms received at the PCC in a full scan time is less than 3 alarms.

3) If the number of alarms exceeds three then reformat the alarms to suit the fact structure called by the AI language. CLIPS requires that all statements should start with a known word before they can be treated as facts. Furthermore, facts of the same category should start with the same word. All alarms will be reformatted to start with the word ' alarm ' in the input list of alarms.

4) Identify the substations where alarms are generated.

5) load the connectivity data tables of the identified substations into the working memory. There is no need to load the data-base for all substations if all alarms are in one or few substations. This partial loading of the data-base will minimize the search space and consequently reduce the reasoning, execution, time of the programs and avoid memory overloading.

6) Identify the type of the alarmed equipment, i.e. buses, lines, transformers,...etc.

7) Load the rules for the identified equipment. Only rules related to the identified equipment types will be loaded. For example, if alarms were generated for lines, only " line out of service " and " line open ended " rules will be loaded. In addition, the general suppression and display rules will be loaded.

8) Initiate the IAP.

Steps 4 through 6 were intended to optimize the IAP reasoning and minimize the search space. However, such requirements imply the partitioning of the IAP knowledge-base rules in a way to allow for selective loading of only some parts as needed. The partitioning is as follows:

a. Rules for lines are built together in 'Line.clp' which includes all rules for "line out of service" and "line open ended" .

b. Rules for "bus de-energized" are built under 'bus.clp '.

c. Rules for "transformer de-energized" are built under 'trans.clp'.

d. Rules for "generator off-line" are built under 'gen.clp '.

e. Rules for "loss of load" are built under ' load.clp '.

f. Rules for "partial black out" are built under ' blk.clp'. Unlike previous partitions, this partition will be loaded if received alarms include at least one alarm for a line and one for a bus in any substation. This criterion was established because the minimum configuration for a substation should necessarily include a line and a bus.

f. Rules for "high voltage" , "low voltage", and other general rules for suppression and display functions are all grouped together under 'rest.clp'. These rules will be loaded every time IAP is initiated because they apply to almost all cases.

Figure 11 depicts the above structure.

3.5.3 COMPLETING THE IAP IMAGE

In previous sections the production rules for the selected disturbance cases and the techniques for the initiation of IAP were discussed. The process of suppression and display of indicative, non-consistent, and unrelated alarms was also

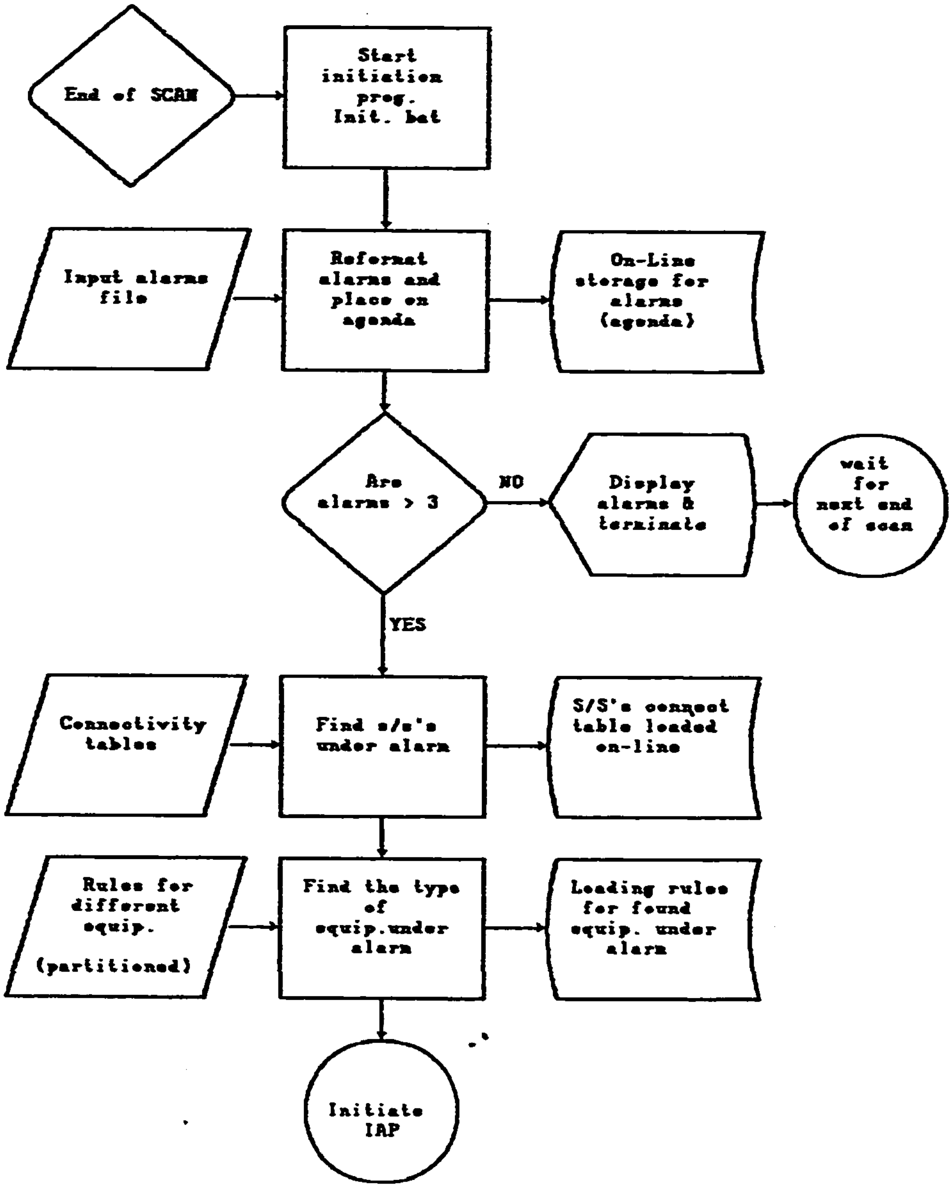


Figure 11: The Initialization Process

discussed. However, these rules must be put together in the most efficient manner to build the knowledge base in order to facilitate the required operation of the IAP. The rules in the knowledge base must be structured and organized according to specific guidelines in order to achieve the IAP goals. These guidelines are as follows.

1) Wrong and contradictory conclusions must be ruled out.

2) The rules for the more frequent disturbances need to be tested first

3) Alarms that might be needed in more than one rule should not be suppressed till the time when they are not needed.

In order to achieve these guidelines, the IAP should fire, execute, the engaged rules in a specified order. A kind of priority assignment is needed for some rules. Such a feature is available in CLIPS and called Salience declaration. It is achieved by adding a new statement to the beginning of any rule in the following manner:

(Declare (Salience number))
Where number can be any integer from 0 to 10,000.

When multiple rules are in the agenda, the rule with the highest priority will fire first. The declaration statement has to be stated as the first pattern in any rule.

The established guidelines imply that the rules must be placed in the following order:

1. Rules governing realization of disturbances must have the highest priority .
2. Rules realizing disturbances must have higher priority than the same rules for incomplete group of alarms.
3. Rules regarding confirmation will be reasoned third, after the last two groups.
4. The rules for suppressing breaker tripping alarms must have lower priority than any confirmation rules.
5. The production rules for the transmission " line out of service " must have higher priority than the rules for " line open ended " and loss of load.
6. The rules for the realization of " partial black out " should come after the rules for " line out of service" and " bus de-energized" and before the rest of other realization rules. Further, rules for the display of any reached conclusion for the buses and the lines must be placed after the rules for " partial black out " . This should avoid the display of any reached conclusion in a substation that is experiencing a black out other than the black out message.

A block diagram that outlines the IAP overall logic is shown in Figure 12. The IAP program is provided in Appendix - C.

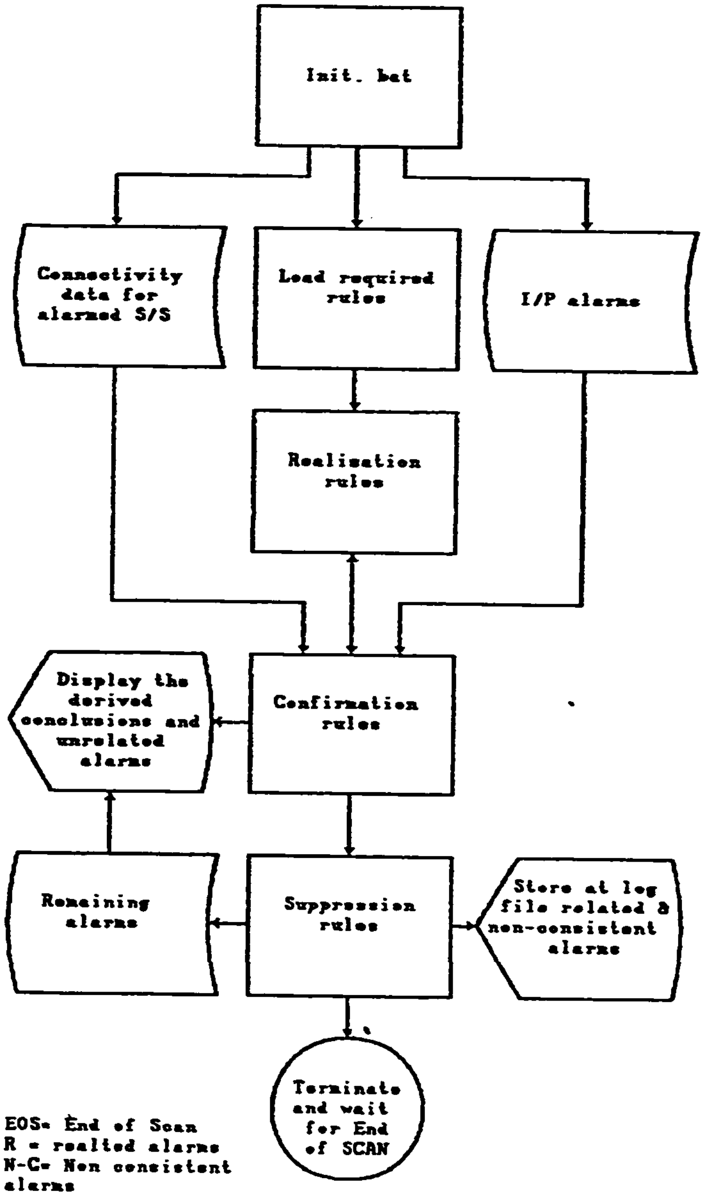


Figure 12: The Overall Operation of the IAP

CHAPTER 4

TEST CASES, RESULTS AND ANALYSIS

4.1 INTRODUCTION

The developed Intelligent Alarm Processor (IAP) is a general type expert system. It can be applied to any power systems model, provided that limits and constraints are met in the selected model.

To demonstrate the operations, capabilities, and limitations of the developed IAP two power system models were experimented. The first model is a three substations system consisting of 3 buses, 1 generator, 1 transformer, 3 lines, 20 breakers and 4 loads. The second model consists of 8 buses, 13 generators, 13 transformer, 19 lines, 7 load lines, and 64 breakers. It represents a portion of SCECO-EAST power system network.

A discussion of the IAP input files together with the naming convention adopted through out this work will be presented first.

4.2 THE IAP SYSTEM ARRANGEMENT AND FILES ORGANIZATION

The IAP system is composed of three components:

1) Initiation Part (Init. bat)- This a batch file that act as a command file. It causes all interactive commands on the system to be executed in a sequence. The operation of this part, which was described in chapter 3, is divided into three major tasks as follows.

a) Testing the received alarms. This task starts by reformatting the received alarms. Then the number of alarms, the substations and the type of equipments generating the alarms are determined. The rules for these tasks are grouped in one file called " decider.clp " . The file is loaded into the system at the beginning of the "init.bat" operation.

b) Loading the substations and rules determined in the previous step. the rules for this task are contained in a file called " selector.clp ".

c) Initialization of the IAP. The IAP will be initialized if the established criteria are met.

The file " selector.clp " includes the names of all substations comprising the modeled network. Consequently, when the IAP is applied to a different model that its substations have different names they should be reflected in " selector.clp ". The other files in " init.bat " do not pertain any information that is local to a specific network. Therefore, they can be used for different power system models without any change.

The filing arrangement in this part is depicted by figure 13.

2) Knowledge Base (KB) Rules - This is the heart of the IAP operation. The rules were built in a general type logic that is not specific to any particular power system model. The rules for each type of equipment were grouped in one CLIPS file . The names for the different equipment types were provided in section 3.5.2 of the last chapter.

3) Connectivity data base - This part requires a total change of data whenever applied to a different model. While the connectivity data base is not part of the IAP, it is attached to it and used extensively by the IAP. The connectivity data for each substation is defined in a file that has the same name as the substation name with a ".clp" extension, i.e. if the substation name is "north", the file name will be "north.clp". Substation file names have to be indicated in the "selector.clp" file.

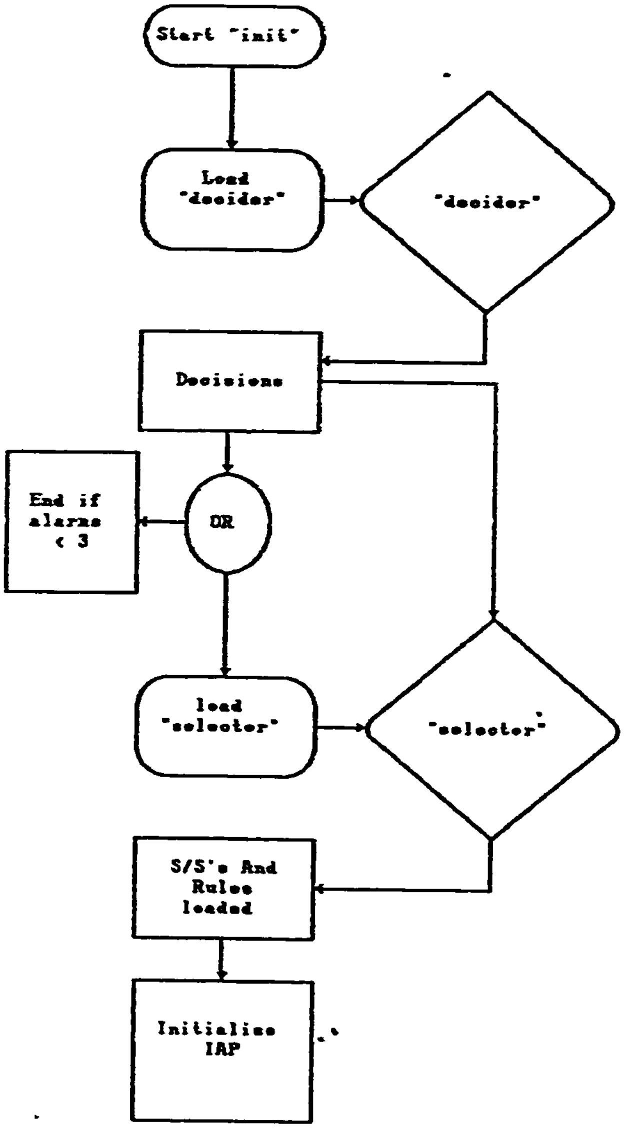


Figure 13: "init.bat" filing arrangement

The input alarms are stored, on the system, in a file named "msg.dat", where ".dat" is the extension designating data type information in CLIPS. All received alarms will be printed on a logger in accumulative order for historical and later use. The logger is simulated by a software file named " log.dat ".

4.3 THE FIRST TEST NETWORK

The first test case is shown in figure 14. it is selected from IEEE Proceedings [25]. Its associated connectivity data are shown in tables 3 and 4. The circuit breaker status data shown in table 6 are not part of the data base but are telemetered values which are subject to change.

In this section, the set of alarms received at the PCC for some selected disturbance cases discussed in chapter 3 will be processed. The capability of the IAP in identifying and verifying each disturbance case will be demonstrated. In addition, the features of presenting conclusions suppressing redundant alarms, and displaying unrelated alarms will be shown.

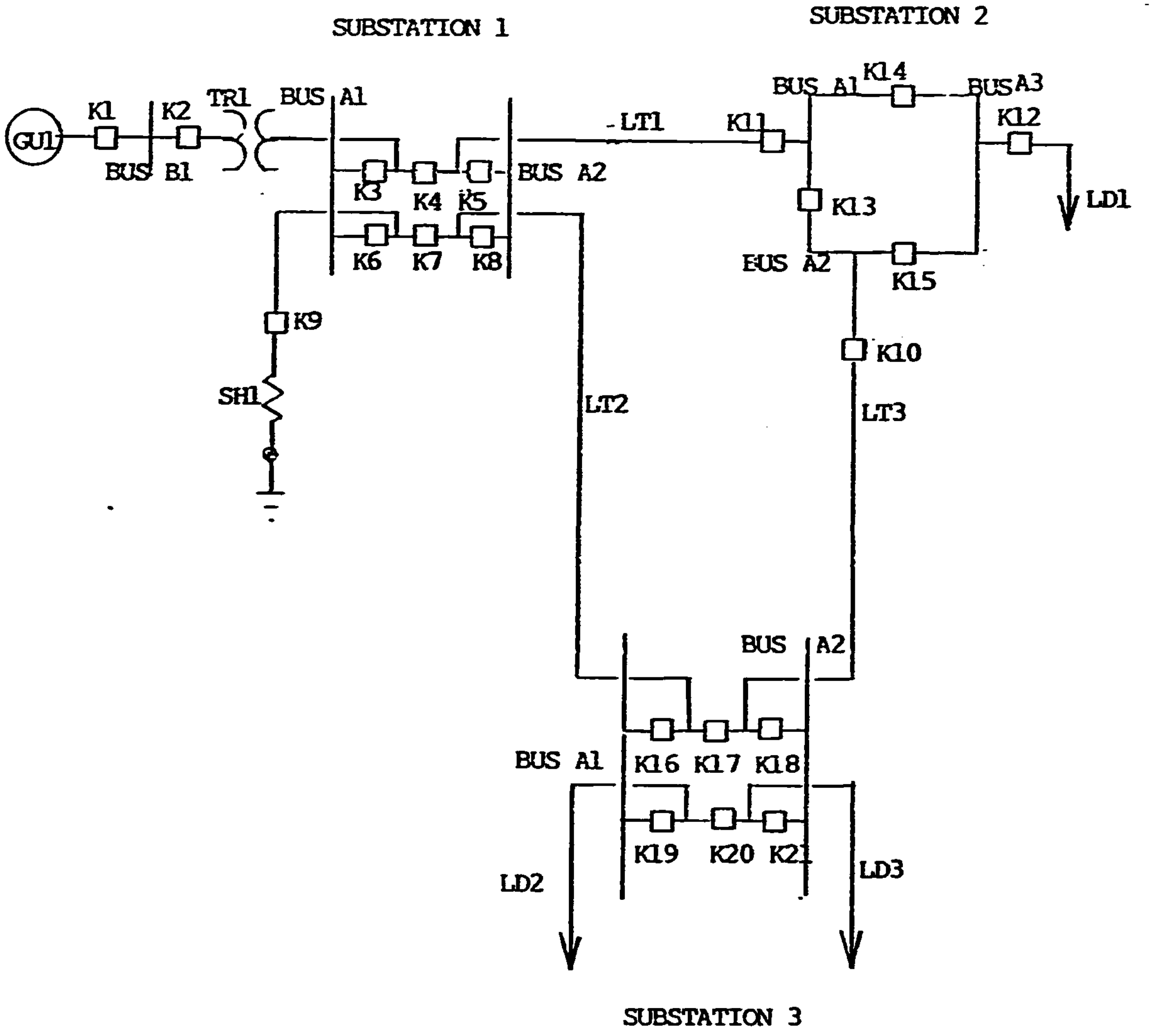


Figure 14 : The line diagram for first test case

TABLE 3. Bus-section connectivity data base for the first test model

Substation Number	No.	Type	Equipment Identification
1	1	Gen. unit	GU1
	2	Bus	B1
	3	Transformer	TR1
	4	Bus	A1
	5	Bus	A2
	6	Transformer	TR1
	7	Line	LT1
	8	Connection	
	9	Line	LT2
	10	Shunt	SH1
2	11	Bus	A1
	12	Bus	A2
	13	Bus	A3
	14	Line	LT1
	15	Line	LT3
	16	Load	LD1
3	17	Bus	A1
	18	Bus	A2
	19	Line	LT2
	20	Line	LT3
	21	Load	LD2
	22	Load	LD3

TABLE 4. Breaker-section connectivity data of the first test model

No.	From Bus Sec	To Bus Sec.	Substation No.	Status
K1	1	2		Close
K2	2	3		Close
K3	4	6		Open
K4	6	7	1	Open
K5	7	5		Close
K6	4	8		Close
K7	8	9		Close
K8	9	5		Close
K9	10	8		Open
K10	12	15		Open
K11	14	11		Close
K12	13	16	2	Close
K13	11	12		Close
K14	11	13		Close
K15	12	13		Close
K16	17	19		Close
K17	19	20		Close
K18	20	18	3	Close
K19	17	21		Close
K20	21	22		Open
K21	22	18		Close

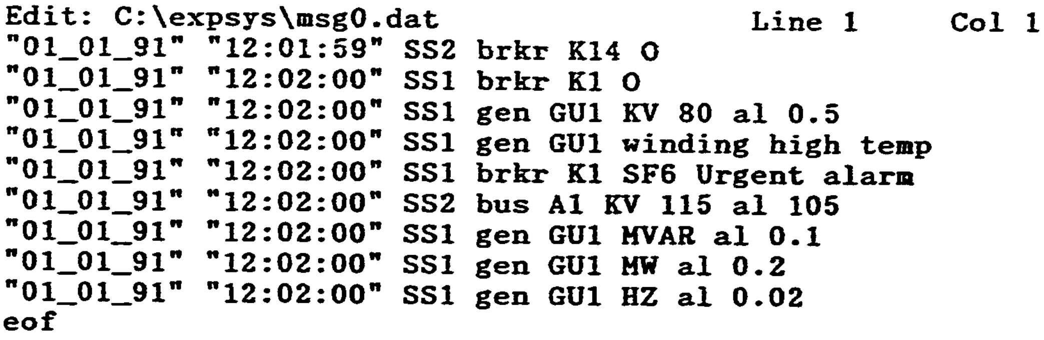
4.3.1 Generator Off-Line

The first disturbance tested is for a generator off-line case. For the network under study the PCC will receive five indicative alarms when "GU1" goes off-line. These alarms are shown in figure 15(a). Additional unrelated and related non-consistent, alarms were added to demonstrate the capability of the IAP to suppress the non-consistent related alarms and to display the unrelated alarms together with the reached conclusion. When this set of alarms is stored in a message file and IAP is initiated, it will produce one intelligent message in addition the unrelated alarms as shown in figure 15(b).

Another test was run to demonstrate the capability of the IAP to diagnose a disturbance case even when some of the indicative alarms for that case are not present. The absence of some indicative alarms could be due to an RTU or communication malfunction. The IAP will indicate to the dispatcher the presence of bad instrumentation. The message file for this scenario together with IAP output are shown in figures 16(a) and 16(b), respectively.

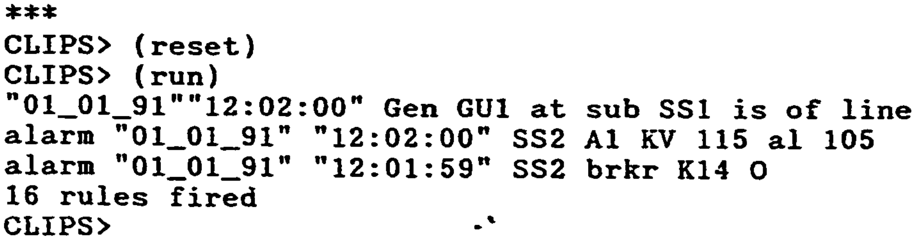
4.3.2 Line out-of-Service

When the line "LT1" between two substations, 1 and 2, goes out-of-service the alarms shown in figure 17(a) are observed at the PCC. Unrelated alarms on breaker



```
Edit: C:\expsys\msg0.dat                       Line 1     Col 1
"01_01_91" "12:01:59" SS2 brkr K14 O
"01_01_91" "12:02:00" SS1 brkr K1 O
"01_01_91" "12:02:00" SS1 gen GU1 KV 80 al 0.5
"01_01_91" "12:02:00" SS1 gen GU1 winding high temp
"01_01_91" "12:02:00" SS1 brkr K1 SF6 Urgent alarm
"01_01_91" "12:02:00" SS2 bus A1 KV 115 al 105
"01_01_91" "12:02:00" SS1 gen GU1 MVAR al 0.1
"01_01_91" "12:02:00" SS1 gen GU1 MW al 0.2
"01_01_91" "12:02:00" SS1 gen GU1 HZ al 0.02
eof
```

Figure 15(a): Input alarms for generator off-line scenario



```
***
CLIPS> (reset)
CLIPS> (run)
"01_01_91""12:02:00" Gen GU1 at sub SS1 is of line
alarm "01_01_91" "12:02:00" SS2 A1 KV 115 al 105
alarm "01_01_91" "12:01:59" SS2 brkr K14 O
16 rules fired
CLIPS>
```

Figure 15(b): The IAP output for the scenario in Figure 15(a)

```
Edit: C:\expsys\msgx0.dat                    Line 1     Col 1
"01_01_90" "12:01:59" SS2 brkr K14 O
"01_01_90" "12:01:59" SS2 brkr K14 SF6 Urgent al
"01_01_90" "12:02:00" SS1 brkr K1 O
"01_01_90" "12:02:00" SS1 bus  A2 60 HZ 59.5
"01_01_90" "12:02:00" SS1 gen GU1 KV 80 al 0.5
"01_01_90" "12:02:00" SS1 gen GU1 High winding temp
"01_01_90" "12:02:00" SS1 gen GU1 MVAR al 0.1
"01_01_90" "12:02:00" SS1 gen GU1 MW al 0.2
eof
```

Figure 16(a): Input alarms for Gen. off-line with some missing alarms

```
***
CLIPS> (reset)
CLIPS> (run)
"01_01_90" "12:02:00" Gen GU1 at SS1 is off line and bad inst
alarm "01_01_90" "12:02:00" SS1 bus A2 60 HZ 59.5
alarm "01_01_90" "12:01:59" SS2 brkr K14 SF6 Urgent al
alarm "01_01_90" "12:01:59" SS2 brkr K14 O
17 rules fired
CLIPS>
```

Figure 16 (b): The output of case in figure 16 (b)

```
Edit: C:\expsys\msgxxx.dat                 Line 1      Col 1
"01_01_90" "12:01:58" SS1 brkr K5 O
"01_01_90" "12:01:58" SS1 bus A2 HZ 230 al 59.8
"01_01_90" "12:01:58" SS1 brkr K4 O
"01_01_90" "12:01:58" SS2 brkr K11 O
"01_01_90" "12:01:58" SS1 line LT1 KV 230 al 0
"01_01_90" "12:01:58" SS1 line LT1 MVAR 230 al 1
"01_01_90" "12:01:58" SS1 line LT1 MW 230 al 0
"01_01_90" "12:01:58" SS2 line LT1 KV 230 al 0
"01_01_90" "12:01:58" SS2 line LT1 MVAR 230 al 0
"01_01_90" "12:01:58" SS2 line LT1 MW 230 al 1.5
"01_01_90" "12:01:59" nmi SS1 line LT1 230 dc
"01_01_90" "12:01:59" nmi SS2 line LT1 230 dc
"01_01_90" "12:01:59" nmi line LT1 230 de
"01_01_90" "12:01:59" SS2 brkr K14 O
"01_01_90" "12:01:58" SS1 brkr K4 SF6 urgent alarm
"01_01_90" "12:01:57" SS1 line LT1 Loss of RS 1
"01_01_90" "12:01:57" SS2 line LT1 Loss of RS 2
eof
```

Figure 17(a): The input alarms for line out of service case

```
***
CLIPS> (reset)
CLIPS> (run)
"01_01_90""12:01:59"line LT1 bet_n sub SS2 & SS1 is out of service
alarm "01_01_90" "12:01:59" SS2 brkr K14 O
alarm "01_01_90" "12:01:58" SS1 bus A2 HZ 230 al 59.79999924
28 rules fired
CLIPS>
```

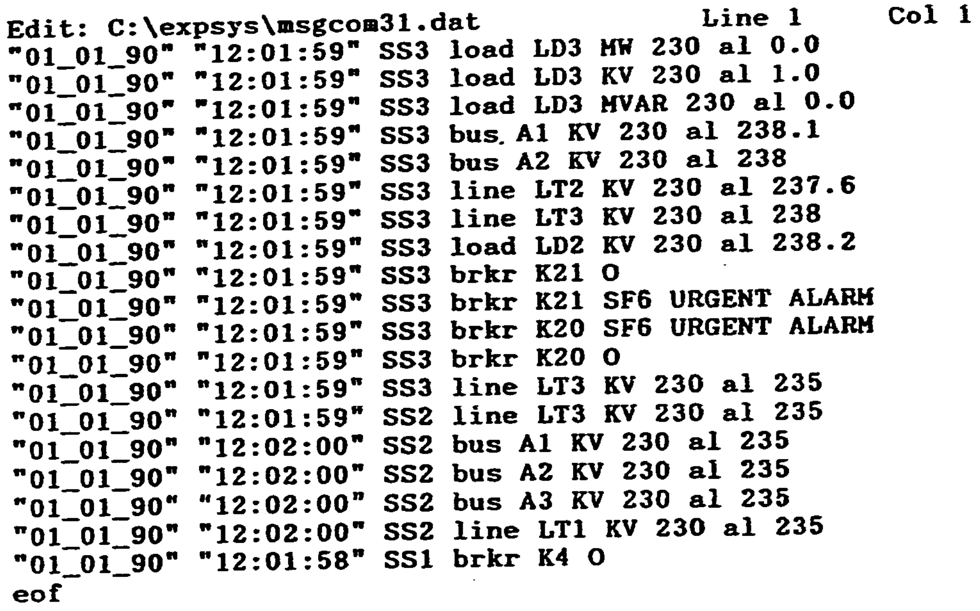
Figure 17(b): The output for alarms in figure 17(a)

14 and for a frequency drop on bus A2 in substation 1 are also present. The dispatcher needs to see these two alarms in addition to the disturbance message. Further, related non-consistent alarms for the "loss of relay set AC/DC " on line "LT1" and "SF6 Urgent alarm" on breaker 11 are also present. These last two alarms must be suppressed since they convey redundant information to the dispatcher. The IAP output for this scenario is shown in figure 17(b). If one or two of the indicative alarms do not appear at the PCC, where it should as a consequent of "line out-of-service", the IAP is able to detect the disturbance and will present the conclusion with a warning about the failure of some telemetry devices.

4.3.3 Loss of Load Causing High Voltage

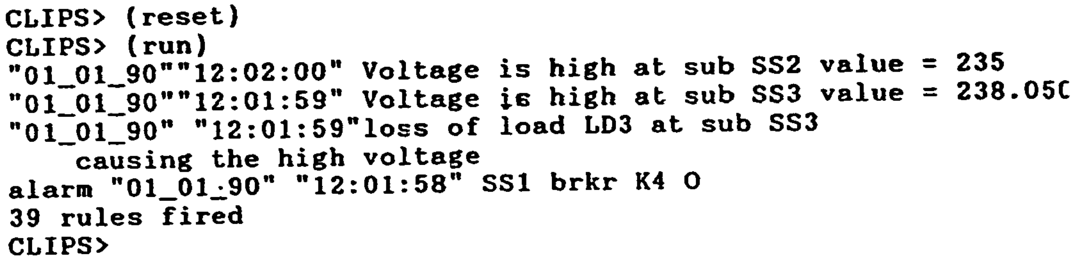
This scenario can be observed when load " LD3 " at substation 3 is dropped from the system causing a high voltage at the substation and at the neighboring substation 2. The alarms for this scenario are shown in figure 18(a). Two non-consistent alarms on breakers K20 and K21 are added to the scenario. In addition, two more unrelated alarms on breaker K4 and bus A1 at substation 1 are also added. The IAP suppresses the first two alarms and displays the later two together with the derived conclusions.

The IAP output for this scenario is depicted in figure 18(b). If one or more of the voltage alarms are not present in the input alarms, same results will be produced but with a special message reflecting the failure of some telemetry devices. The input



```
Edit: C:\expsys\msgcom31.dat                 Line 1      Col 1
"01_01_90" "12:01:59" SS3 load LD3 MW 230 al 0.0
"01_01_90" "12:01:59" SS3 load LD3 KV 230 al 1.0
"01_01_90" "12:01:59" SS3 load LD3 MVAR 230 al 0.0
"01_01_90" "12:01:59" SS3 bus A1 KV 230 al 238.1
"01_01_90" "12:01:59" SS3 bus A2 KV 230 al 238
"01_01_90" "12:01:59" SS3 line LT2 KV 230 al 237.6
"01_01_90" "12:01:59" SS3 line LT3 KV 230 al 238
"01_01_90" "12:01:59" SS3 load LD2 KV 230 al 238.2
"01_01_90" "12:01:59" SS3 brkr K21 O
"01_01_90" "12:01:59" SS3 brkr K21 SF6 URGENT ALARM
"01_01_90" "12:01:59" SS3 brkr K20 SF6 URGENT ALARM
"01_01_90" "12:01:59" SS3 brkr K20 O
"01_01_90" "12:01:59" SS3 line LT3 KV 230 al 235
"01_01_90" "12:01:59" SS2 line LT3 KV 230 al 235
"01_01_90" "12:02:00" SS2 bus A1 KV 230 al 235
"01_01_90" "12:02:00" SS2 bus A2 KV 230 al 235
"01_01_90" "12:02:00" SS2 bus A3 KV 230 al 235
"01_01_90" "12:02:00" SS2 line LT1 KV 230 al 235
"01_01_90" "12:01:58" SS1 brkr K4 O
eof
```

Figure 18(a): The input alarms for loss of load scenario



```
CLIPS> (reset)
CLIPS> (run)
"01_01_90""12:02:00" Voltage is high at sub SS2 value = 235
"01_01_90""12:01:59" Voltage is high at sub SS3 value = 238.050
"01_01_90" "12:01:59"loss of load LD3 at sub SS3
    causing the high voltage
alarm "01_01_90" "12:01:58" SS1 brkr K4 O
39 rules fired
CLIPS>
```

Figure 18(b): The output of the IAP for the scenario shown in figure 18(a) alarms for this case is shown in figure 19(a). The output for this new scenario is depicted in figure 19(b).

4.3.4 Transformer De-energized and Substation Black Out

This case is encountered when the transformer "TR1" is substation 1is de-energized. It can be represented by the alarms shown in figure 20(a). It is assumed that the de-energization of " TR1" causes the islanding of the whole substation. Further, two non-consistent events occurred simultaneously at substation 2 which are the tripping of breaker "K14" the drop in bus "A3" frequency. The dispatcher needs to know the last two alarms in addition to the black out occurrence. The rest of alarms at substation 1 should be suppressed. The IAP output for this case is shown if figure 20(b).

4.4 THE SECOND TEST NETWORK

So far, The IAP was tested for some of the major disturbance cases.It was assumed that these cases will happen in isolation of the others. The occurrence of multiple disturbances at the same time and the wide spread of alarms in the network is another task the IAP should handle. To test the performance of the IAP for system wide disturbances and to further demonstrate the IAP capabilities, a bigger network

```
"01_01_90" "12:01:59" SS3 load LD3 MW 230 al 0.0
"01_01_90" "12:01:59" SS3 load LD3 KV 230 al 1.0
"01_01_90" "12:01:59" SS3 load LD3 MVAR 230 al 0.0
"01_01_90" "12:01:59" SS3 bus A2 KV 230 al 238
"01_01_90" "12:01:59" SS3 brkr K21 O
"01_01_90" "12:01:59" SS3 brkr K21 SF6 URGENT ALARM
"01_01_90" "12:01:59" SS3 brkr K20 SF6 URGENT ALARM
"01_01_90" "12:01:59" SS3 brkr K20 O
"01_01_90" "12:01:59" SS3 line LT3 KV 230 al 235
"01_01_90" "12:01:59" SS2 line LT3 KV 230 al 235
"01_01_90" "12:02:00" SS2 bus A1 KV 230 al 235
"01_01_90" "12:02:00" SS2 bus A2 KV 230 al 235
"01_01_90" "12:02:00" SS2 bus A3 KV 230 al 235
"01_01_90" "12:02:00" SS2 line LT1 KV 230 al 235
"01_01_90" "12:01:58" SS1 brkr K4 O
eof
```

Figure 19 (a): The input alarms for loss of load scenario with some missing alarms

```
CLIPS> (reset)
CLIPS> (run)
"01_01_90""12:02:00" Voltage is high at sub SS2 value = 235.350
"01_01_90""12:01:59" KV alarm in SS SS3 but not at
 all eq.'s value = 235.19999695
 suggesting that some telemetry are not
 operational in SS SS3
"01_01_90" "12:01:59"loss of load LD3 at sub SS3
    causing the high voltage
alarm "01_01_90" "12:01:58" SS1 brkr K4 O
31 rules fired
CLIPS>
```

Figure 19 (b): The output for the scanario in figure 19 (a)

```
"01_01_90" "12:01:58" SS2 bus A3 HZ 230 al 59.5
"01_01_90" "12:01:58" SS1 brkr K4 O
"01_01_90" "12:01:58" SS1 brkr K3 O
"01_01_90" "12:01:58" SS1 brkr K2 O
"01_01_90" "12:02:58" SS1 trans TR1 LSKV 40 al 0.1
"01_01_90" "12:02:58" SS1 trans TR1 HSKV 230 al 1.1
"01_01_90" "12:02:58" SS1 trans TR1 MW al 0.2
"01_01_90" "12:02:58" SS1 trans TR1 MVAR al 0.3
"01_01_90" "12:02:00" SS1 gen GU1 MW al 0.2
"01_01_90" "12:01:58" SS1 brkr K5 O
"01_01_90" "12:01:58" SS2 brkr K14 O
"01_01_90" "12:01:58" SS1 line LT1 KV 230 al 2
"01_01_90" "12:01:58" SS1 line LT1 MVAR 230 al 1
"01_01_90" "12:01:58" SS1 line LT1 MW 230 al 0
"01_01_90" "12:01:59" nmi SS1 line LT1 230 dc
"01_01_90" "12:01:58" SS1 brkr K8 O
"01_01_90" "12:01:58" SS1 brkr K7 O
"01_01_90" "12:01:58" SS1 brkr K6 O
"01_01_90" "12:01:58" SS1 line LT2 KV 230 al 2
"01_01_90" "12:01:58" SS1 line LT2 MVAR 230 al 1
"01_01_90" "12:01:58" SS1 line LT2 MW 230 al 0
"01_01_90" "12:01:59" nmi SS1 line LT2 230 dc
"01_01_90" "12:01:58" SS1 bus A2 KV 230  al 1
"01_01_90" "12:01:58" SS1 bus A2 HZ 230 al 0
"01_01_90" "12:01:58" SS1 brkr K5 SF6 urgent alarm
eof
```

Figure 20(a): The input alarms for transformer de-energized causing a black out in the substation

```
****
CLIPS> (reset)
CLIPS> (run)
 ** Black out at SS1 **
alarm "01_01_90" "12:01:58" SS2 brkr K14 O
alarm "01_01_90" "12:01:58" SS2 bus A3 HZ 230 al 59.5
46 rules fired
CLIPS>
```

Figure 20(b): The IAP output for the case shown in figure 20 (a) than the one used above was selected. The data of the network were provided in section 4.1. The network line diagram is shown in figure 21 and the bus-section and breakers-section connectivity tables for the four substations are shown in tables 5 and 6, respectively.

4.4.1 Substation Black-out

The first test case was for a black out of a substation. Substation JADWIP is fed by two lines Jad WIP1 and Jad WIP2. When these two lines are disconnected substation JadWIP will experience a black out. The alarms which will be received at PCC are shown in figure 22(a). In spite that the two feeding lines will be out of service in addition to buses de-energization at JadWIP, the IAP should display a higher level message indicating that the substation experienced a black out. All other alarms and messages in the substation will be suppressed. The output by the IAP is shown in figure 22(b).

The capabilities of IAP is further illustrated by presenting a different scenario for the black out. The scenario assumed that one of the two lines feeding substation "JAD WIP 1", line "Jad WIP 2", was already on scheduled outage when the other line became out-of-service. Alarms will be produced for one line only and other equipment at the substation. No alarms will be observed for line "JAD WIP 2" in this

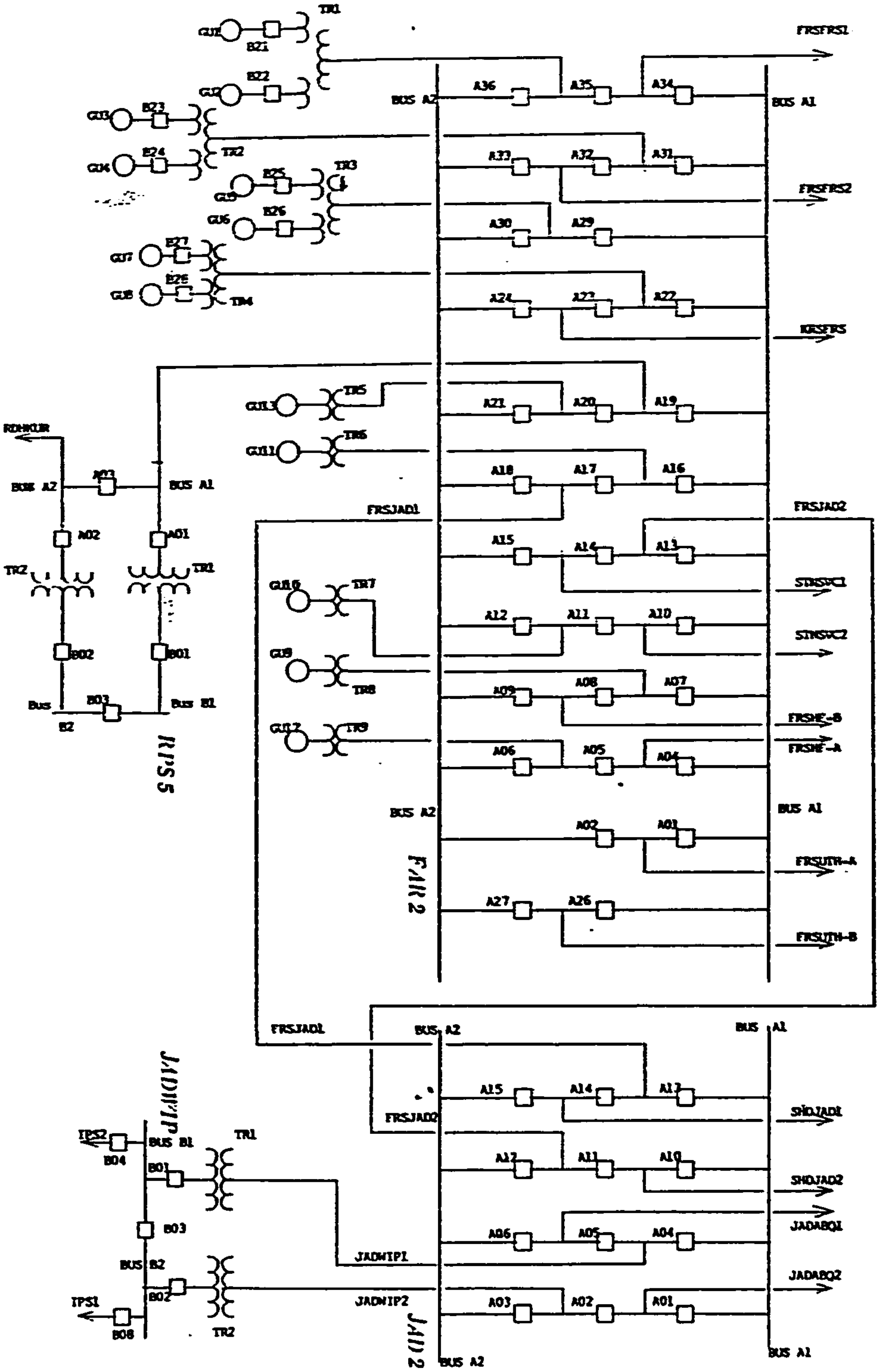


Figure 21: The line diagram of second test case

TABLE 5. Bus-Section Data for second test network

Substation	Equip No.	Type	Equip I.D.
FARAS 230 (FAR2)	01	gen	GU1
	02	gen	GU2
	03	gen	GU3
	04	gen	GU4
	05	gen	GU5
	06	gen	GU6
	07	gen	GU7
	08	gen	GU8
	09	trans	TR1
	10	trans	TR1
	11	trans	TR2
	12	trans	TR2
	13	trans	TR3
	14	trans	TR3
	15	trans	TR4
	16	trans	TR4
	17	trans	TR4
	18	trans	TR2
	19	trans	TR1
	20	trans	TR3
	21	gen	GU13
	22	gen	GU11
	23	trans	TR5
	24	trans	TR6
	25	gen	GU10
	26	gen	GU09
	27	gen	GU12
	28	trans	TR7
	29	trans	TR8
	30	trans	TR9
	31	bus	A1
	32	bus	A2
	33	line	FRSFRS1
	34	line	FRSFRS2
	35	line	KRSFRS
	36	line	FRSRPS5
	37	line	FRSJAD1
	38	load	STNSVC1
	39	line	FRSJAD2

continued on p.139

Continuation of Table 5

Substation	Equip No.	Type	Equip I.D.
FARAS 230 (FAR2) cont'd	40	load	STNSVC2
	41	line	FRSHF_B
	42	line	FRSHF_A
	43	line	FRSUTH_A
	44	line	FRSUTH_B
JADIDAH 230 (JAD2)	53	bus	A1
	54	bus	A2
	55	line	FRSJAD1
	56	line	SHDJAD1
	57	line	FRSJAD2
	58	line	SHDJAD2
	59	line	JADWIP1
	60	load	JADABQ1
	61	load	JADABQ2
	62	line	JADWIP2
RPS5	45	line	FRSRPS5
	46	load	RDHKUR
	47	trans	TR1
	48	trans	TR2
	49	trans	TR1
	50	trans	TR2
	51	bus	B1
	52	bus	B2
JADWIP	63	line	JADWIP2
	64	line	JADWIP1
	65	trans	TR2
	66	trans	TR1
	67	bus	B2
	68	bus	B1
	69	load	IPS1
	70	load	IPS2

TABLE 6. Circuit Breaker Section for second test network

Substation	Breaker No.	From	To	Status
FARAS 230 (FAR2)	B21	1	9	Close
	B22	2	10	Close
	B23	3	11	Close
	B24	4	12	Close
	B25	5	13	Open
	B26	6	14	Close
	B27	7	15	Open
	B28	8	16	Close
	A1	31	43	Open
	A2	32	43	Close
	A4	31	42	Close
	A5	30	42	Close
	A6	32	30	Open
	A7	31	29	Close
	A8	29	41	Close
	A9	32	41	Open
	A10	31	40	Close
	A11	40	28	Close
	A12	32	28	Close
	A13	31	39	Close
	A14	39	38	Close
	A15	32	38	Open
	A16	31	24	Close
	A17	24	37	Close
	A18	37	32	Close
	A19	31	36	Close
	A20	36	23	Close
	A21	32	23	Close
	A22	31	17	Close
	A23	17	35	Open
	A24	35	32	Close
	A26	31	44	Close
	A27	44	32	Close
	A29	31	20	Close
	A30	20	32	Close
	A31	31	18	Close
	A32	18	34	Close
	A33	34	32	Close

continued on p.141

Continuation of Table 6

Substation	Breaker No.	From	To	Status
FARAS 230	A34	33	31	Open
(FAR2) cont'd	A35	19	33	Close
	A36	19	32	Close
JADIDAH 230 (JAD2)	A1	53	61	Close
	A2	61	62	Close
	A3	54	62	Close
	A4	53	59	Open
	A5	59	60	Close
	A6	54	60	Close
	A10	53	58	Open
	A11	57	58	Open
	A12	54	57	Open
	A13	53	55	Close
	A14	55	56	Close
	A15	54	56	Close
RPS5	A1	45	47	Close
	A2	46	48	Close
	A3	45	46	Open
	B1	49	51	Close
	B2	50	52	Close
	B3	51	52	Close
JADWIP	B1	66	68	Close
	B2	65	67	Close
	B3	67	68	Close
	B4	68	70	Open
	B8	67	69	Open

```
"01_01_90" "12:01:58" JAD2 brkr A2 O
"01_01_90" "12:01:58" JAD2 brkr A4 O
"01_01_90" "12:01:58" JAD2 brkr A5 O
"01_01_90" "12:01:58" JAD2 line JADWIP1 KV 230 al 0
"01_01_90" "12:01:58" JAD2 line JADWIP1 MVAR 230 al 1
"01_01_90" "12:01:58" JAD2 line JADWIP1 MW 230 al 0
"01_01_90" "12:01:58" JADWIP line JADWIP1 KV 230 al 0
"01_01_90" "12:01:58" JADWIP line JADWIP1 MVAR 230 al 1
"01_01_90" "12:01:58" JADWIP line JADWIP1 MW 230 al 0
"01_01_90" "12:01:59" nmi JADWIP line JADWIP1 230 dc
"01_01_90" "12:01:59" nmi line JADWIP1 230 de
"01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm
"01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1
"01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2
"01_01_90" "12:01:58" JADWIP brkr B1 O
"01_01_90" "12:02:58" JADWIP trans TR1 LSKV 34.5 al 0.1
"01_01_90" "12:02:58" JADWIP trans TR1 HSKV 230 al 1.1
"01_01_90" "12:02:58" JADWIP trans TR1 MW al 0.2
"01_01_90" "12:02:58" JADWIP trans TR1 MVAR al 0.3
"01_01_90" "12:01:59" nmi JAD2 line JADWIP1 230 dc
"01_01_90" "12:01:58" JADWIP bus B2 KV 34.5  al .3
"01_01_90" "12:01:58" JADWIP brkr B3 O
"01_01_90" "12:01:58" JADWIP brkr B8 O
"01_01_90" "12:01:58" JADWIP bus B2 HZ 34.5 al 0
"01_01_90" "12:01:58" JADWIP brkr B3 SF6 urgent alarm
"01_01_90" "12:01:58" JADWIP bus B2 SF6 urgent alarm
"01_01_90" "12:01:58" JAD2 line JADWIP2 KV 230 al 0
"01_01_90" "12:01:58" JAD2 line JADWIP2 MVAR 230 al 1
"01_01_90" "12:01:58" JAD2 line JADWIP2 MW 230 al 0
"01_01_90" "12:01:58" JADWIP line JADWIP2 KV 230 al 0
"01_01_90" "12:01:58" JADWIP line JADWIP2 MVAR 230 al 1
"01_01_90" "12:01:58" JADWIP line JADWIP2 MW 230 al 0
"01_01_90" "12:01:59" nmi JADWIP line JADWIP2 230 dc
"01_01_90" "12:01:59" nmi line JADWIP2 230 de
"01_01_90" "12:01:59" nmi JAD2 line JADWIP2 230 dc
eof
```

Figure 22(a):The input alarms for a black out case with all lines de-energized

```
****
CLIPS> (reset)
CLIPS> (run)
 ** Black out at JADWIP **
"01_01_90""12:01:59"line JADWIP1 bet_n sub JADWIP & JAD2 is out of service
"01_01_90""12:01:59"line JADWIP2 bet_n sub JADWIP & JAD2 is out of service
56 rules fired
CLIPS>
```

Figure 22(b): The IAP output for the case shown in figure 22(a)

case. The IAP should find out that the two lines are not feeding the substation anymore and that the substation is under black out case. Figure 23(a) gives the input alarms for this scenario, while table 23(b) gives the output messages as predicted by the IAP.

4.4.2 Wide System Disturbance

This case is intended to demonstrate the capabilities of the IAP to detect the occurrence of multiple disturbances on the network at more than one substation. A scenario is composed such that bus "A2" at "JAD 2" is de-energized, line "FRSJAD2" is out of service, line "JAD WIP 1" disconnected, transformer "TR1" at "Jad WIP" and load "JAD ABQ 1" is lost at "JAD 2" substation. The alarms that would be received at the PCC for such a scenario are given in figure 24(a). The IAP will suppress all alarms relevant to these disturbances, except the alarms that are related to the network configuration or are related to equipment other than these that experienced disturbances. For instance, frequency or voltage changes at other disturbances. Some other alarms related to equipment that failed, like SF6 alarms on breaker or transformer, will be suppressed. The IAP output is shown in figure 24(b).

```
"01_01_90" "12:01:58" JAD2 brkr A4 O
"01_01_90" "12:01:58" JADWIP line JADWIP1 KV 230 al 0
"01_01_90" "12:01:58" JADWIP line JADWIP1 MVAR 230 al 1
"01_01_90" "12:01:58" JADWIP line JADWIP1 MW 230 al 0
"01_01_90" "12:01:59" nmi JADWIP line JADWIP1 230 dc
"01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm
"01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1
"01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2
"01_01_90" "12:01:58" JADWIP brkr B1 O
"01_01_90" "12:02:58" JADWIP trans TR1 LSKV 34.5 al 0.1
"01_01_90" "12:02:58" JADWIP trans TR1 HSKV 230 al 1.1
"01_01_90" "12:02:58" JADWIP trans TR1 MW al 0.2
"01_01_90" "12:02:58" JADWIP trans TR1 MVAR al 0.3
"01_01_90" "12:01:58" JADWIP bus B1 KV 34.5  al .3
"01_01_90" "12:01:58" JADWIP brkr B3 O
"01_01_90" "12:01:58" JADWIP bus B1 HZ 34.5 al 0
"01_01_90" "12:01:58" JADWIP brkr B3 SF6 urgent alarm
"01_01_90" "12:01:58" JADWIP bus B2 SF6 urgent alarm
eof
```

Figure 23(a): The input alarms for a black out case with some lines de-energized

```
****
CLIPS> (reset)
CLIPS> (run)
 ** Black out at JADWIP **
alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2
alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1
alarm "01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm
alarm "01_01_90" "12:01:58" JAD2 brkr A4 O
35 rules fired
CLIPS>
```

Figure 23(b): The IAP output for the case shown in figure 23(a)

```
"01_01_90" "12:01:58" FAR2 brkr A13 O
"01_01_90" "12:01:58" FAR2 brkr A14 O
"01_01_90" "12:01:58" FAR2 bus A2 HZ 230 al 60.5
"01_01_90" "12:01:58" FAR2 bus A2 KV 230 al 235
"01_01_90" "12:01:58" JAD2 brkr A11 O
"01_01_90" "12:01:58" JAD2 brkr A12 O
"01_01_90" "12:01:58" FAR2 line FRSJAD2 KV 230 al 0
"01_01_90" "12:01:58" FAR2 line FRSJAD2 MVAR 230 al 1
"01_01_90" "12:01:58" FAR2 line FRSJAD2 MW 230 al 0
"01_01_90" "12:01:58" JAD2 line FRSJAD2 KV 230 al 0
"01_01_90" "12:01:58" JAD2 line FRSJAD2 MVAR 230 al 0
"01_01_90" "12:01:58" JAD2 line FRSJAD2 MW 230 al 1.5
"01_01_90" "12:01:59" nmi FAR2 line FRSJAD2 230 dc
"01_01_90" "12:01:59" nmi JAD2 line FRSJAD2 230 dc
"01_01_90" "12:01:59" nmi line FRSJAD2 230 de
"01_01_90" "12:01:58" FAR2 brkr A14 SF6 urgent alarm
"01_01_90" "12:01:57" FAR2 line FRSJAD2 Loss of RS 1
"01_01_90" "12:01:57" JAD2 line FRSJAD2 Loss of RS 1
"01_01_90" "12:01:58" JAD2 bus A2 KV 230  al 1
"01_01_90" "12:01:58" JAD2 brkr A15 O
"01_01_90" "12:01:58" JAD2 brkr A3 O
"01_01_90" "12:01:58" JAD2 brkr A6 O
"01_01_90" "12:01:58" JAD2 bus A2 HZ 230 al 0
"01_01_90" "12:01:58" JAD2 brkr A15 SF6 urgent alarm
"01_01_90" "12:01:58" JAD2 bus A2 SF6 urgent alarm
"01_01_90" "12:01:59" JAD2 load JADABQ1 MW 230 al 0.0
"01_01_90" "12:01:59" JAD2 load JADABQ1 KV 230 al 0.0
"01_01_90" "12:01:59" JAD2 load JADABQ1 MVAR 230 al 0
"01_01_90" "12:01:59" JAD2 load JADABQ1 loss of RS1
"01_01_90" "12:01:59" JAD2 load JADABQ1 loss of RS2
"01_01_90" "12:01:59" JAD2 brkr A5 O
"01_01_90" "12:01:59" JAD2 brkr A4 O
"01_01_90" "12:01:58" JAD2 line JADWIP1 KV 230 al 0
"01_01_90" "12:01:58" JAD2 line JADWIP1 MVAR 230 al 1
"01_01_90" "12:01:58" JAD2 line JADWIP1 MW 230 al 0
"01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm
"01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1
"01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2
"01_01_90" "12:01:58" JADWIP brkr B1 O
"01_01_90" "12:02:58" JADWIP trans TR1 LSKV 34.5 al 0.1
"01_01_90" "12:02:58" JADWIP trans TR1 HSKV 230 al 1.1
"01_01_90" "12:02:58" JADWIP trans TR1 MW al 0.2
"01_01_90" "12:02:58" JADWIP trans TR1 MVAR al 0.3
"01_01_90" "12:01:59" nmi JAD2 line JADWIP1 230 dc
eof
```

Figure 24(a): The input alarms for wide system dusturbance

```
****
CLIPS> (reset)
CLIPS> (run)
"01_01_90""12:01:58" sub JAD2 BUS A2 230 KV is deenergized
"01_01_90""12:01:59"line FRSJAD2 bet_n sub JAD2 & FAR2 is out of service
"01_01_90""12:01:59" line JADWIP1 open ended at sub JAD2
"01_01_90""12:02:58" Trans TR1 at sub JADWIP is deenergized
"01_01_90""12:01:58" KV alarm in SS FAR2 but not at all equip.'s value = 234
 suggesting that some telemetry are not
 operational in SS FAR2
"01_01_90" "12:01:59" loss of load JADABQ1 at sub JAD2
alarm "01_01_90" "12:01:58" FAR2 bus A2 HZ 230 al 60.5
82 rules fired
CLIPS>
```

Figure 24(b): The IAP output for the case shown in figure 24(a)

4.4.3 The Historical File

In real time environment, all received alarms are printed on a logger, for historical or diagnosis purposes at later time. They are printed in an accumulating order, i.e. new alarms will be printed after the last alarms received in the previous scan. Therefore, the "log.dat" which was built to simulate the PCC logger should append all new alarms to the already existing alarms every time IAP is initiated. Figure 25 shows all alarms received in three different scans, assuming that the three disturbance scenarios presented above were received by the PCC in three different consecutive scans. These were extracted from "log.dat" file.

4.5 CONCLUSION REMARKS

The example cases discussed in this chapter demonstrated the flexibility, efficiency, and the ability of the IAP to handle various combination of disturbances. The developed IAP could perform the following functions :-

1. Identify any one or a combination of the nine disturbance cases discussed in this thesis. It sends special messages to the display, suppress all alarms that the dispatcher does not need to see, and

```
2211
"alarm "01_01_90" "12:01:58" JAD2 brkr A4 0"
"alarm "01_01_90" "12:01:58" JAD2 brkr A5 0"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 MW 230 al 0"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP1 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP1 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP1 MW 230 al 0"
"alarm "01_01_90" "12:01:59" nmi JADWIP line JADWIP1 230 dc"
"alarm "01_01_90" "12:01:59" nmi line JADWIP1 230 de"
"alarm "01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2"
"alarm "01_01_90" "12:01:58" JADWIP brkr B1 0"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 LSKV 34.5 al 0.1"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 HSKV 230 al 1.1"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 MW al 0.2"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 MVAR al 0.3"
"alarm "01_01_90" "12:01:59" nmi JAD2 line JADWIP1 230 dc"
"alarm "01_01_90" "12:01:58" JADWIP bus B2 KV 34.5  al .3"
"alarm "01_01_90" "12:01:58" JADWIP brkr B3 0"
"alarm "01_01_90" "12:01:58" JADWIP brkr B8 0"
"alarm "01_01_90" "12:01:58" JADWIP bus B2 HZ 34.5 al 0"
"alarm "01_01_90" "12:01:58" JADWIP brkr B3 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JADWIP bus B2 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP2 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP2 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP2 MW 230 al 0"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP2 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP2 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP2 MW 230 al 0"
"alarm "01_01_90" "12:01:59" nmi JADWIP line JADWIP2 230 dc"
"alarm "01_01_90" "12:01:59" nmi line JADWIP2 230 de"
"alarm "01_01_90" "12:01:59" nmi JAD2 line JADWIP2 230 dc"
"alarm "01_01_90" "12:01:58" JAD2 brkr A4 0"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP1 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP1 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" JADWIP line JADWIP1 MW 230 al 0"
"alarm "01_01_90" "12:01:59" nmi JADWIP line JADWIP1 230 dc"
"alarm "01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2"
"alarm "01_01_90" "12:01:58" JADWIP brkr B1 0"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 LSKV 34.5 al 0.1"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 HSKV 230 al 1.1"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 MW al 0.2"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 MVAR al 0.3"
"alarm "01_01_90" "12:01:58" JADWIP bus B1 KV 34.5  al .3"
"alarm "01_01_90" "12:01:58" JADWIP brkr B3 0"
"alarm "01_01_90" "12:01:58" JADWIP bus B1 HZ 34.5 al 0"
"alarm "01_01_90" "12:01:58" JADWIP brkr B3 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JADWIP bus B2 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" FAR2 brkr A13 0"
"alarm "01_01_90" "12:01:58" FAR2 brkr A14 0"
"alarm "01_01_90" "12:01:58" FAR2 bus A2 HZ 230 al 60.5"
"alarm "01_01_90" "12:01:58" FAR2 bus A2 KV 230 al 234"
"alarm "01_01_90" "12:01:58" JAD2 brkr A11 0"
"alarm "01_01_90" "12:01:58" JAD2 brkr A12 0"
"alarm "01_01_90" "12:01:58" FAR2 line FRSJAD2 KV 230 al 0"
"alarm "01_01_90" "12:01:5R" FAR2 line FRSJAD2 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" FAR2 line FRSJAD2 MW 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 line FRSJAD2 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 line FRSJAD2 MVAR 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 line FRSJAD2 MW 230 al 1.5"
"alarm "01_01_90" "12:01:59" nmi FAR2 line FRSJAD2 230 dc"
"alarm "01_01_90" "12:01:59" nmi JAD2 line FRSJAD2 230 dc"
"alarm "01_01_90" "12:01:59" nmi line FRSJAD2 230 de"
"alarm "01_01_90" "12:01:58" FAR2 brkr A14 SF6 urgent alarm"
"alarm "01_01_90" "12:01:57" FAR2 line FRSJAD2 Loss of RS 1 "
"alarm "01_01_90" "12:01:57" JAD2 line FRSJAD2 Loss of RS 1"
"alarm "01_01_90" "12:01:58" JAD2 bus A2 KV 230  al 1"
"alarm "01_01_90" "12:01:58" JAD2 brkr A15 0"
"alarm "01_01_90" "12:01:58" JAD2 brkr A3 0"
"alarm "01_01_90" "12:01:58" JAD2 brkr A6 0"
"alarm "01_01_90" "12:01:58" JAD2 bus A2 HZ 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 brkr A15 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JAD2 bus A2 SF6 urgent alarm"
"alarm "01_01_90" "12:01:59" JAD2 load JADABQI MW 230 al 0.0"
"alarm "01_01_90" "12:01:59" JAD2 load JADABQI KV 230 al 0.0"
"alarm "01_01_90" "12:01:59" JAD2 load JADABQI MVAR 230 al 0"
"alarm "01_01_90" "12:01:59" JAD2 load JADABQI loss of RS1"
"alarm "01_01_90" "12:01:59" JAD2 load JADABQI loss of RS2"
"alarm "01_01_90" "12:01:59" JAD2 brkr A5 0"
"alarm "01_01_90" "12:01:59" JAD2 brkr A4 0"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 KV 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 MVAR 230 al 1"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 MW 230 al 0"
"alarm "01_01_90" "12:01:58" JAD2 brkr A4 SF6 urgent alarm"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 1"
"alarm "01_01_90" "12:01:58" JAD2 line JADWIP1 Loss of RS 2"
"alarm "01_01_90" "12:01:58" JADWIP brkr B1 0"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 LSKV 34.5 al 0.1"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 HSKV 230 al 1.1"
"alarm "01_01_90" "12:02:58" JADWIP trans TR1 MW al 0.2"
"alarm "01 01 90" "12:02:58" JADWIP
```

Figure 25: The historical file output (LOG.DAT)

displays other alarms that either the dispatcher need to know about or are not related to the disturbances.

2. IAP will still recognize disturbances even when some of the indicative alarms are missing. It will send a special message containing information about the disturbance and will point out the possibility of telemetry device failure at the disturbed substation.
3. The IAP can handle substation black out and wide-spread disturbances, provided that the disturbances are of the nine defined types, even when they occur in the same scan period.
4. The possibility of reaching a wrong conclusion is minimized because IAP is designed to display all received alarms when it fails to draw a conclusion.
5. The developed IAP is flexible. Any additional knowledge can be amended to the knowledge base easily.
6. IAP is designed to load the data of the affected substations only. This makes a better utilization of the computer working memory.
7. The IAP can handle different power systems. Only the system data need to be modified.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 SUMMARY

The application of artificial intelligence to power systems alarm processing has been studied. A review of both expert systems techniques and power systems alarms at power control centers was made. The power system alarm types, conditions, classifications, and generations were analyzed. The expert system shell CLIPS is used to build a rule-based expert system that process the power systems alarms in an intelligent manner.

The developed expert system, called Intelligent Alarm Processor (IAP), was designed to emulate the power system dispatcher's way of thinking during a power system disturbance. It analyzes the power system alarms received at the Power Control Center (PCC) to make conclusions as to what have happened on the network. The knowledge base of the IAP was built on established rules and conditions that were derived from literature and through questioning experienced power dispatchers. The IAP performs two prime functions:

1. Reduces the number of alarms received at the power control center considerably to a fewer number of intelligent messages (alarms). This feature is best realized when the power system network experiences a disturbance or abnormality.

2. Detects faulty telemetry devices installed on the power system network components to monitor analog values. The detection is realized when the PCC expects alarms from those components, during a power system disturbance, and it fails to acquire them.

In developing the IAP some of the power system major disturbances were first identified. The alarms that usually accompany each case were defined and empirical rules for each disturbance were then formulated. Finally, a prototype was built. It connects these production rules and other supporting rules, defines the input-output relations, and facilitates the interaction between the rules and the data files on the system.

The connectivity data in the power system data base played an essential role in the reasoning process of the IAP. It supplied knowledge to the IAP about the different power system components and their connectivity relations.

The IAP was tested for two power system models. The first is a small model that was used to debug the IAP. The second model is a bigger size network. It was extracted from the SCECO East network. Several scenarios were assumed for each model. Each scenario simulated a different power system disturbance on the network. The IAP performed as expected. In some cases it reduced the number of received alarms, by more than 80% of the total received alarms. While producing the intelligent messages, the IAP will keeps all received alarms in a log file for historical and future reference purposes.

5.2 CONCLUSION

The application of the intelligent alarm processor to the alarm processing in modern Power Control Centers (PCC) will certainly enhance the overall operation of the PCC. The IAP major achievement will be reducing the time required for the dispatcher to analyze the received alarms. This will assist him in doing faster restoration actions by identifying what have happened on the network. Such an assistance is best realized when the power system experiences an abnormality or a major disturbance.

Another achievement of the IAP is detecting faulty telemetry devices that are installed on the various components of the power system. This will reduce the risk of making wrong assumptions on the state of the network.

Such enhancements do not only provide assistance to the power system dispatcher, but also minimize the risks on the integrity of the power system network. This is due to the fact that some of the power system faults are of cascading nature and their effect can be minimized or eliminated by fast restoration action.

5.3 CONTRIBUTION OF THE THESIS

The main contribution of the thesis is the problem formulation and solution of the excessive alarms at the power control center during major disturbances on the power network. A prototype expert system is developed for alarm processing. It emulates the dispatcher's way of deriving conclusions form the received alarms. It utilizes the system connectivity data and built-in knowledge to produce concise and short explanation of the cause of these alarms.

5.3 RECOMMENDATIONS FOR FUTURE WORK

The application of artificial intelligence to power system operations is a new field that is still under research cycle. The field still lacks the methods by which such an application can be implemented in real-time environments of the power control center. Consequently, there is a wide area of possible enhancements and new applications. Several aspects of needed future work to expand the developed IAP and to enhance its functions are discussed below:

1. Development of an IAP with perspective type messages. The IAP, developed in this work, provides diagnostic type messages as to what has happened on the network. Prescriptive type messages would further suggest what need to be done to return the network to its normal state.

2. Use the developed IAP on a power system simulator. This will detect any shortcomings and will enhance the possibility of taking it one further step; to a real power system application. This will also identify any additional knowledge that may be needed.

3. Study the possibility of developing local IAP at substation RTU software. The local IAP will produce higher intelligent messages than what current RTU's are producing now related to each power system component in the substation. Then a central IAP, to be installed at the PCC software, can be upgraded to deal with intelligent messages and will be capable of producing global type messages of higher level of intelligence. The global type messages will describe the general condition and state of the whole over power system network during disturbances and abnormalities.

APPENDICES

APPENDIX - A

TABLE A.1 Differences Between Artificial Intelligence Programming And Conventional Programming

Procedural Languages	AI Languages
- (BASIC, FORTRAN, PASCAL).	- (PROLOG, LISP).
- Use algorithms to solve problems.	- Use heuristic to solve problems.
- Numerically addressed.	- Symbolically structured.
- Most efficient at numerical processing.	- Most efficient at formal reasoning.
- Systems created and maintained by programmers.	- Systems developed and maintained by knowledge engineer.
- Use structured programming.	- Interactive and cyclic development.
- Database.	- Knowledge-base in a global common.

TABLE A.2 Tools For Expert Systems

NAME	EXPERT	OPS5	KEE	KES	CLIPS
Manufacturer	Elis & Kulikowski, Dept. of Comp Sci, Rutgers Univ., New Brunswick, NJ	Dept. of Comp. Sci Carnegie-Mellon Univ.	Intellicorp. 707 Laurel St. Menlo Park, CA	Software A&E 1500 Wilson Blvd. Arlington, VA	AI section/NASA Johnson Space Center
Introduced	1981	1980	1983	1983	1987
Consultation Paradigm	Diagnosis/Prescription	Planning and Diagnosis / Prescription	Hybrid Tool	Diagnosis/Prescription	Diagnosis/Prescription
Knowledge Rep., Inference, and	IF-THEN Confidence factor	IF-THEN rules recognize act cycle with conflict resolution	Frame with slots object-oriented IF-THEN rules,user defined inference	IF-THEN rules multiple objects, inheritance Procedural control Bayesian probabilities	IF- THEN rules recognize act cycle with conflict resolution data driven system
Implementation	FORTRAN	Fronze LISP,Mac LISP, VAX 11/780	LISP/Common LISP Xerox 1100S,Symbolics 3600, LMI LAMBDA,TI Explorer	A-LISP, Wisconsin LISP Franz LISP. DEC VAX/VMS VAX/UNIX, CDC CYBER and APOLLO Xerox 1100's IBM PC with (IQLISP)	LISP, C Language DEC VAX/VMS DEC C & FORTRAN Compiler HP 9000 (HP-UX UNIX) and IBM PC

Cont'd on page 157

Continuation of Table A.2 Tools For Expert Systems

NAME	EXPERT	OPS5	KEE	KES	CLIPS
User/KE	Line-oriented questionnaire	Line oriented	Interactive,grap-pical displays of the knowledge base	Line oriented display Explanation (explain) Trace and probes inter-face to outside data base	Line-oriented Both interactive and works as embedded ES
Applications	Several field Systems Serum Protein/Diagnosis prig. Rheumatic Disease prig./ Expert Lag Analysis System	Major fielded systems including CON, SELL	Several Fielded Systems GENESIS	-	-
Support	-	Manual	Product include 3 day workshop and 10 days on-sigh consultation	Manual	Manual

APPENDIX - B

QUESTIONNAIRE FOR POWER SYSTEMS DISPATCHERS ON INTELLIGENT ALARM PROCESSOR

MAZEN A. SALAH
31/08/1990

Q1. LIST THE POWER SYSTEM DISTURBANCES THAT WHEN OCCURRED ON THE POWER SYSTEM NETWORK GENERATES MANY ALARMS (i.e. line out of service, loss of major load,etc...)?.

1. Transmission LINES Tripping
2. Power transformer Tripping
3. GENErator tripping
4. Reactor tripping
5. Loss of Big LOAD.
6. System voltage fluctuation.
7. System Hz fluctuation
8.

Q2. FOR EACH OF THE DISTURBANCES YOU HAVE MENTIONED ABOVE, WHAT ARE THE ASSOCIATED ALARMS THAT ARE DISPLAYED IN THE SUBSTATION (OR SUBSTATIONS) UNDER DISTURBANCE AND NEIGHBORING SUBSTATIONS ?

1. a Change of BRK status. OPEN/Close etc.
b. LINE out of Service.
c. LINE OPEN END.
d Fault Recorder trigger.
e Associate lines, XFMR,s limits MW+MVARs cross the limits
f. " STN. Bus voltage limits cross the limits Low/High

g. STN: Loss of SCAN. (i.e. CHANN: through transmission lines PLC.)

2. a XFMR BRK STN: Changed open.
   b. Bus tie Close on Auto. STATUS to Close.
   c. Change in Bus voltage.
   d. T/Recorder trigger.
   f. Associate transformer LOAD. MW/MVAR if limits cross the value.
   g. Urgent / Non urgent alarm on transformer. BRK. etc.

3. a. Small Generator cause almost ~~no~~ less alarm mainly change of BRK STATUS.

   b. in Case of big. GEN. Loss. Big change in Freq. originate many alarm. not only the O/P Bus. Hz. almost all Hz Point in the Area.
   c. voltage Low/Hi cross the limits
   d. IN Case of U/Freq. all alarms in Case of #1.

4\. a Change of link STATUS Breaker.
b. All Buses voltage level which
Cross Hi Lo limits.
c. Hi/Low VARR on the Generators

5: a Hi: System Freq. lot of alarm @ all STN.
b. AGC alarm DES: GEN Low/Hi
c. System voltage alarm as stated above.
d. Can cause loss Some GEN: on
Hi. Freq. mostly alarm: on AGC.

6\. a. Un Balance System
b. Fault fail to clear in time.
Lot of alarm Low/Hi Voltages. on
Bus BAR as well on Generator.
MVAR Low/Hi. etc.

7 almost Same as #6.

Q3. SUPPOSE THAT YOU RECEIVED MANY ALARMS SUDDENLY, WHAT ARE THE FIRST THINGS THAT YOU LOOK AT ? WHICH ALARMS ARE THE MOST PECULIAR AND DECISIVE THAT YOU WILL CHECK FOR THERE PRESENCE FIRST TO DERIVE YOUR CONCLUSIONS?.

1. all Priority #1 alarm mostly RED.
2 Change of BRK STATUS.
3 MAIN BOARD LINES STATUS. 4. System HZ.
9. LINE. TRIPPED OR OPEN END
Review all BRK tripped. CHECK. S.O.E.
alarm. to localize the fault.

Q4. IF THERE ARE SEVERAL POWER SYSTEMS DISTURBANCES ON THE NETWORK, CAN YOU RANK THE PRIORITY IN WHICH YOU WOULD LIKE THEM TO BE DISPLAYED (ASSUME THE DISTURBANCES THAT WERE DEFINED IN THE FIRST QUESTION).

IN Case of Several Disturbance. Priority always to Save
the Rest of System in hand.

Q5. FOR EACH CASE EXPLAINED ABOVE (IN QUESTIONS 1 & 2) WHAT MESSAGES DO YOU THING SHOULD BE DISPLAYED AND WHAT SHOULD BE SUPPRESSED (NOT SHOWN)?

for line tripping BRK status open +
line out of service or open END is Energi.
fault Recorders trigger alarm.
Same for XFMR. trip. BRK status.
all the alarms which listed in
Q. 2. list according to the Priority. [illegible]
first. 3 + 4 alarm should be
displayed Rest can be suppressed

APPENDIX - C

LISTING OF THE IAP PROTOTYPE

This appendix provides a listing of the program for the knowledge base prototype in CLIPS language. It is divided into the following major parts :

I. The Initialization Process. This part includes the following sections:
   - The decider
   - The selector
   - The final section for saving all received alarms

II. The Knowledge base rules. This part includes all the empirical rules for the major disturbances discussed in chapter 3 of this thesis. The rules for each case were ordered in the following sequence:
   - The realization rules for the complete group of alarms.
   - The realization rules for the special cases where some alarms are missing
   - The confirmation rules
   - The suppression and display rules

III. The rules for displaying alarms where no conclusion was reached

```
;*********  THE INITIALIZATION PROCESS *****************

(load "decider.clp")
(load "selector.clp")
; selector loads sceco substations
;(load "selectr2.clp")
; select1 loads IEEE substations
(reset)
(run)
(facts)
(excise start)
(excise count)
(excise det2)
(excise det3)
(excise det4)
(excise det5)
(excise det6)
(excise det7)
(load "test.clp")
(load "rest.clp")
(reset)
(run)

;*************** THE DECIDER *********************
; The read and reformat rules
(defrule start
  ?int <- (initial-fact)
=>
  (retract ?int)
  (open "msgcom2.dat" msg1 "r")
  (open "log.dat" log "a")
  (open "log1.dat" log1 "w")
  (bind ?k (readline msg1))
  (bind ?k1 (str_cat "alarm " ?k))
  (str_assert ?k1)
  (fprintout log ?k1 crlf)
  (assert(count1 1))
  (assert(init ver_ss)))

; The rules for counting and identifying the substations under
  alarm

(defrule count
  (declare (salience 1080))
  ?cnt <- (count1 ?n)
  ?int <-(init ver_ss $?ss)
  ?alm <-(alarm $?flds)
=>
 (fprintout log1 "alarm " $?flds crlf)
 (retract ?cnt ?int ?alm)
 ; msg*.dat includes the alarms
```

```
  (bind ?k (readline msgl))
 (if(&&(neq ?k "EOF")(neq ?k "eof"))
  then
  (bind ?k1 (str_cat "alarm " ?k))
  (str_assert ?k1)
  (fprintout log ?k1 crlf)
  (assert(count1 =(+?n 1)))
  ; It is assummed that 4th element is the sub name
  (bind ?c(nth 3 $?flds))
  (bind ?s(member ?c $?ss))
  (if(and(= ?s 0)
        (neq ?c nmi))
    then(fprintout t "?c is " ?c " " ?s crlf)
      (assert(init ver_ss $?ss ?c))
        (assert(getnow ?c))
    else(assert(init ver_ss $?ss)))
   else
   (fprintout logl "EOF" crlf)
   (close)
   (if(<= ?n 3)
   ; the creteria set if alarms are less
        than 3 program should not run
   ; where n is the number of alarms
   then
   (fprintout t "alarms do not worth running the IAP "crlf)
   else
   (fprintout t "There are  " ?n
     " Alarms on the network in " $?ss crlf))))
; The rules for determining the type of equipments under alarm
(defrule det2
 (declare (salience 1090))
 (not(loading l_d_e))
 (alarm $? line $?)
=>
 (assert(loading l_d_e )))

(defrule det3
 (declare (salience 1087))
 (not(loading trans_f))
 (alarm $? trans $?)
=>
; trans. rules will be loaded if alarms on any trans. >/ 2.
  (assert(loading trans_f)))

(defrule det4
 (declare (salience 1085))
 (not(loading gen_f))
 (alarm $? gen $?)
=>
; gen rules will be loaded if alarms on any gen are >/ 2.
  (assert(loading gen_f)))
```

```
(defrule det5
 (declare (salience 1083))
 (not(loading load_ls))
 (alarm $? load $?)
=>
; loss of load rules will be loaded if alarms
: on any load are >/ 2.
  (assert(loading load_ls)))

(defrule det6
 (declare (salience 1082))
 (not(loading bus_de))
 (alarm $? bus $?)
=>
  (assert(loading bus_de)))

(defrule det7
 (declare (salience 1081))
 (not(loading blk_out))
 (loading bus_de)
 (loading l_d_e)
=>
  (assert(loading blk_out)))

;**************** THE SELECTOR ******************
; The rules for loading the connectivity data for the
substations under alarm and the rules for the alarmed
equipment
(defrule test1
?f<-(getnow FAR2)
=>
 (retract ?f)
 (load "far2.clp")
(defrule test2
?f<-(getnow JAD2)
=>
 (retract ?f)
 (load "jad2.clp")
(defrule test3
?f<-(getnow RPS5)
=>
 (retract ?f)
 (load "rps5.clp")
(defrule test4
?f<-(getnow JADWIP)
=>
 (retract ?f)
 (load "jadwip.clp")
(defrule decide11
(declare (salience 1726))
 (loading l_d_c)
=>
```

```
 (load "lindc.clp"))

(defrule decide22
(declare (salience 1025))
(loading l_d_e)
=>
 (load "lindc.clp")
 (load "linde.clp"))

(defrule decide33
(declare (salience 1024))
(loading trans_f)
=>
  (load "trans.clp"))

(defrule decide44
(declare (salience 1023))
(loading gen_f)
=>
  (load "gen.clp"))

(defrule decide55
(declare (salience 1022))
(loading load_ls)
=>
  (load "load.clp"))

(defrule decide66
(declare (salience 1021))
(loading bus_de)
=>
  (load "busde.clp"))

(defrule decide77
(declare (salience 1020))
(loading blk_out)
=>
 (load "blkout.clp"))

;************ SAVING ALL RECEIVED ALARMS **********

(defrule init11
(declare (salience 1650))
?f<- (initial-fact)
=>
 (retract ?f)
 (open "logl.dat" logl "r")
 (bind ?k(readline logl))
 (str_assert ?k)
 (if(neq ?k "EOF")
  then
 (assert(initial-fact))
```

```
 else(close log1)))

;************** THE KNOWLEDGE BASE RULES **************

;******  RULES FOR BUS DE-ENERGIZED ******

(defrule bus_de
  (declare (salience 850))
  ?a <- (alarm ?date ?time ?SS bus ?ID KV ?n.val al ?val)
  (test(<= ?val (/ ?n.val 100)))
  ?b <- (alarm ? ? ?SS bus ?ID HZ ?n.val al ?Hzval&:(<=?Hzval
1))
  =>
  (retract ?a ?b)
  (assert(bus_de_inf ?date ?time ?n.val))
  (assert(bus_de ?ID at ?SS)))

(defrule bs_de_bt_bdinst
  (declare (salience 848))
  (or(and ?k <- (alarm ?date ?time ?SS bus ?ID KV
            ?n.val al ?Kv_val)
      (test(<= ?Kv_val (/ ?n.val 100))))
     ?k <- (alarm ?date ?time ?SS bus ?ID HZ ?n.val al
            ?Hz_val&:(<=?Hz_val 1)))
=>
  (assert(bus_de_inf ?date ?time ?n.val))
  (retract ?k)
  (assert(bus_de bd_inst ?ID at ?SS)))

(defrule bus_de_prt1
  (declare (salience 815))
   (bus_de $?g ?ID at ?SS)
   (not(blk_out ?SS))
 ?f<-(bus_de_inf ?d ?t ?n.val)
=>
 (retract ?f)
 (if(eq $?g bd_inst)
 then
(fprintout t ?d ?t " sub " ?SS " BUS " ?ID " " ?n.val
           " KV is deenergized" crlf)
(fprintout t " and bad inst. is detected" crlf)
else(fprintout t ?d ?t " sub " ?SS " BUS " ?ID " " ?n.val
           " KV is deenergized" crlf)))

; The suppression part
(defrule al_remov_bde
   (declare (salience 300))
  (bus_de $? ?ID at ?SS)
  ?r <- (alarm ? ? ?SS brkr ?no $?)
   (dt_bs_sc ?SS ?con bus ?ID)
  (or(dt_br ?SS ?no ?con ? 0)
   (dt_br ?SS ?no ? ?con 0))
```

```
=>
   (retract ?r))

(defrule clean_others_on_busde.
(declare (salience 845))
  (bus_de $? ?ID at ?SS)
 ?f<-(alarm ? ?  ?SS bus ?ID $?)
=>
    (retract ?f))

; ******** Rules For the Generator off line case **********

(defrule gen_of
   (declare (salience 700))
  ?y <- (alarm ?date ?time ?SS gen ?ID KV ?n.val al ?Kv.val)
     (test(<=?Kv.val  (/ ?n.val 100)))
  ?z <- (alarm  ? ? ?SS gen ?ID MW al ?Mw.val&:(<=?Mw.val 2))
  ?w <- (alarm  ? ? ?SS gen ?ID MVAR al ?Mv.val&:(<=?Mv.val
2))
  ?u <- (alarm ? ? ?SS gen ?ID HZ al ?Hz.val&:(<=?Hz.val 1))
=>
   (retract ?y ?z ?w ?u)
   (assert(gen_of ?ID at ?SS))
   (fprintout t ?date ?time " Gen " ?ID " at sub " ?SS
          " is of line" crlf))

(defrule gen_of_bd_ins
  (declare (salience 690))
 ?r <- (alarm ?date ?time ?SS brkr ?brkr_no 0)
      (dt_br ?SS ?brkr_no ?from ?to 0)
      (dt_bs_sc ?SS ?con_no&?from|?to gen ?ID)
     (not(gen_of ?ID at ?SS))
     (or(and(alarm ? ? ?SS gen ?ID KV ?KV_nom al ?KV_val)
         (test(> ?KV_val (/ ?KV_nom 100))))
     (not(alarm ? ? ?SS gen ?ID KV ?KV_nom al ?KV_val))
     (not(alarm ? ? ?SS gen ?ID MW al ?MW_val&:(<=?MW_val 2)))
     (not(alarm ? ? ?SS gen ?ID MVAR al ?MV_val&:(<=?MV_val
2)))
     (not(alarm ? ? ?SS gen ?ID HZ al ?Hz_val&:(<=?Hz_val
1))))
     (not(gen_of bd_inst ?ID at ?SS))
=>
     (retract ?r)
     (fprintout t ?date " " ?time " Gen " ?ID " at " ?SS
            " is off line and bad inst" crlf)
     (assert(gen_of ?ID of_bd_ins at ?SS)))

 (defrule al_remov_gof
   (declare (salience 689))
   (gen_of ?ID $? at ?SS)
   (or ?r<-(alarm ? ? ?SS gen ?ID $?)
    (and ?r <- (alarm ? ? ?SS brkr ?no $?)
```

```
  (dt_bs_sc ?SS ?con gen ?ID)
  (dt_br ?SS ?no $? ?con $?)))
=>
   (retract ?r))

; *** Rules Definig Transformer de-energized case *******

(defrule trans_de
   (declare (salience 600))
  ?y <- (alarm ?date ?time ?SS trans ?ID LSKV
?n.val1 al ?LKV_val)
     (test(<= ?LKV_val (/ ?n.val1 100)))
  ?z <- (alarm ? ? ?SS trans ?ID HSKV ?n.val2 al ?HKV_val)
     (test(<= ?HKV_val (/ ?n.val2 100)))
  ?t <- (alarm ? ? ?SS trans ?ID MW al ?valmw&:(<=?valmw 2))
  ?u <- (alarm ? ? ?SS trans ?ID MVAR al ?valmv&:(<=?valmv 2))
 =>
   (retract ?y ?z ?t ?u)
   (assert(trns_de ?ID at ?SS))
   (fprintout t ?date ?time " Trans " ?ID " at sub " ?SS
             " is deenergized "crlf))

(defrule trans_de_bd_ins
  (declare (salience 590))
  (or(and(alarm ?date ?time ?SS trans ?ID LSKV ?n.val al ?val)
     (test(< ?val (/ ?n.val 100))))
  (and(alarm ?date ?time ?SS trans ?ID HSKV ?n.valu al ?valu)
  (test(< ?valu (/ ?n.valu 100))))
  (alarm ?date ?time ?SS trans ?ID MW al ?valmw&:(<=?valmw 2))
  (alarm ?date ?time ?SS trans ?ID MVAR al ?valmv&:(<=?valmv
2)))
  (not(trns_de $? ?ID at ?SS))
=>
 (fprintout t ?date " " ?time "Trans " ?ID " at " ?SS
              " is DE and/or bad inst" crlf)
 (assert(trans bd_inst ?ID at ?SS)))

(defrule al_remov_tde
   (declare (salience 588))
  (trns_de $? ?ID at ?SS)
  (or ?wr<- (alarm ? ? ?SS trans ?ID $?)
  (and ?wr<- (alarm ? ? ?SS brkr ?No $?)
   (dt_bs_sc ?SS ?con trans ?ID)
   (dt_br ?SS ?No $? ?con $?)))
=>
  (retract ?wr))

; ******* Rules Defining line out of service case ******

(defrule l_o_s
  (declare (salience 840))
  ?a  <- (alarm ?dat ?tim nmi ?SS1 line ?ID ?n.val dc)
```

```
  ?a0 <- (alarm $? nmi ?SS2 line ?ID ?n.val dc)
  ?a1 <- (alarm $? nmi line ?ID ?n.val de)
  ?a2 <- (alarm $? ?SS1 line ?ID KV ?n.val al ?Kv_val1)
  (test(<= ?Kv_val1 (/ ?n.val 100)))
  ?a3 <- (alarm $? ?SS1 line ?ID MW ?n.val al
          ?Mw_val1&:(<=?Mw_val1 2))
  ?a4 <- (alarm $? ?SS1 line ?ID MVAR ?n.val al
         ?Mv_val1&:(<=?Mv_val1 2))
  ?a5 <- (alarm $? ?SS2&~?SS1 line ?ID KV ?n.val al ?Kv_val)
  (test(<= ?Kv_val (/ ?n.val 100)))
  ?a6 <- (alarm $? ?SS2&~?SS1 line ?ID MW ?n.val al
         ?Mw_val2&:(<=?Mw_val2 2))
  ?a7 <- (alarm $? ?SS2&~?SS1 line ?ID MVAR ?n.val al
          ?Mv_val2&:(<=?Mv_val2 2))
=>
  (assert(l_o_s ?ID ?SS1 ?SS2))
  (assert(l.o.s_inf ?ID ?dat ?tim ?n.val))
  (retract ?a ?a0 ?a1 ?a2 ?a3 ?a4 ?a5 ?a6 ?a7))

(defrule los_but_bad_ins
  (declare (salience 838))
  ?a  <- (alarm ?dat ?tim nmi ?SS1 line ?ID ?n.val dc)
  ?a0 <- (alarm $? nmi ?SS2&~?SS1 line ?ID ?n.val dc)
  ?a1 <- (alarm $? nmi line ?ID ?n.val de)
=>
  (assert(l_o_s bd_inst ?ID ?SS1 ?SS2))
  (assert(l.o.s_inf ?ID ?dat ?tim ?n.val))
  (retract ?a ?a0 ?a1))

(defrule line_os_prt
  (declare (salience 813))
  ?f<-(l.o.s_inf ?ID ?date ?time ?n.val)
  (l_o_s ?ID ?SS1 ?SS2)
=>
 (retract ?f)
 (fprintout t ?date ?time "line " ?ID " bet_n sub " ?SS1 " &
  " ?SS2 " is out of service" crlf))

(defrule prt_lin_de_bd
  (declare (salience 808))
  (l_o_s bd_inst ?ID ?SS1 ?SS2)
  ?f<-(l.o.s_inf ?ID ?d ?t ?n.val)
=>
 (retract ?f)


(fprintout t ?d ?t "line " ?ID " is os but bad
instrum. bet_n " ?SS1 " and " ?SS2 crlf))

(defrule clean_alarms_on_l.o.s
 (declare (salience 807))
   (l_o_s $? ?ID ?SS1 ?SS2)
```

```
   ?b<- (alarm ? ?  ?SS&?SS1¦?SS2 line ?ID $?)
=>
  (retract ?b))

(defrule al_remov_los
   (declare (salience 298))
   (l_o_s $? ?ID ?SS1 ?SS2)
 ?r <-(alarm ? ? ?SS&?SS1¦?SS2 brkr ?brkr $?)
   (dt_bs_sc ?SS&?SS2|?SS1 ?con line ?ID)
  (or (dt_br ?SS&?SS2|?SS1 ?brkr ?con ? 0)
      (dt_br ?SS&?SS2|?SS1 ?brkr ? ?con 0))
=>
   (retract ?r))


;** This module finds if there are any line open ended
; under the condition that the line is not out of service ***

(defrule line_dc_prt
  (declare (salience 812))
 ?f<-(ldc_inf ?ID ?d ?t ?n.val)
     (line_dc $?h ?ID at ?SS)
 (not(blk_out ?SS))
=>
  (retract ?f)
  (if(eq $?h bd_inst)
   then
(fprintout t ?d ?t " line " ?ID  " is open ended at sub " ?SS
        " and bad inst is detected" crlf)
 else(fprintout t ?d ?t " line " ?ID
      " open ended at sub " ?SS crlf)))

(defrule line_dc
        (declare (salience 830))
    ?a1  <- (alarm ?dat ?tim nmi ?SS line ?ID ?n.val dc)
    ?a2 <- (alarm ? ? ?SS line ?ID KV ?n.val al ?Kv_val)
    (test(<= ?Kv_val (/ ?n.val 100)))
    ?a3 <- (alarm ? ? ?SS line ?ID MW ?n.val al
           ?Mw_val&:(<=?Mw_val 2))
    ?a4 <- (alarm ? ? ?SS line ?ID MVAR ?n.val al
               ?Mv_val&:(<=?Mv_val 2))
=>
   (assert(line_dc ?ID at ?SS))
   (assert(ldc_inf ?ID ?dat ?tim ?n.val))
   (retract ?a1 ?a2 ?a3 ?a4))

(defrule line_dc_bd_inst
        (declare (salience 825))
    ?a<-(alarm ?dat ?tim nmi ?SS line ?ID ?n.val dc)
   ?a2 <-(alarm ? ? ?SS line ?ID ? ?n.val $?)
=>
  (retract ?a ?a2)
```

```
    (assert(ldc_inf ?ID ?dat ?tim ?n.val))
   (assert(line_dc bd_inst ?ID at ?SS)))

(defrule al_remov_ldc
    (declare (salience 805))
 (line_dc $? ?ID at ?SS)
 (or ?r<- (alarm ? ?  ?SS ? ?ID $?)
  (and ?r <- (alarm ? ? ?SS brkr ?brkr_no $?)
   (dt_bs_sc ?SS ?con_no line ?ID)
   (dt_br ?SS ?brkr_no $? ?con_no $? 0)))
=>
  (retract ?r))

; ****** Rules Defining partial black out case *****

(defrule start_blk_out
 (declare (salience 824))
 (bus_de $? ?ID1 at ?SS)
 (not(blk_out1 ?SS ?))
 (or(line_dc $? ?ID at ?SS)
 (l_o_s $? ?ID ?SS ?)
 (l_o_s $? ?ID ? ?SS))
 =>
  (assert(checkcount 0))
  (assert(chknext 1))
  (assert(blk_out1 ?SS ?ID)))

(defrule black_out2
 (declare (salience 823))
 (blk_out1 ?SS ?ID)
 (dt_bs_sc ?SS ?con line ?ID2&~?ID)
 (not(blk_out ?SS))
 ?f<-(checkcount ?n)
 (not(line ?ID2 skip))
 (not(l_o_s $? ?ID2 ? ?SS))
 (not(l_o_s $? ?ID2 ?SS ?))
 (not(line_dc $? ?ID2 at ?SS))
=>
  (retract ?f)
  (assert(checkcount =(+?n 1)))
  (assert(line ?ID2 skip))
  (assert(other_lines ?ID2 ?con)))

(defrule black_out3
 (declare (salience 822))
 (blk_out1 ?SS ?ID)
 ?f1<-(other_lines ?ID2 ?con)
 (not(blk_out ?SS))
 (dt_br ?SS $? ?con $?)
 (not(dt_br ?SS $? ?con $? C))
=>
 (retract ?f1)
```

```
 (assert(other_line ?ID ?con)))

(defrule black_out4
 (declare (salience 821))
 (blk_out1 ?SS ?ID)
 (checkcount 0)
=>
  (assert(blk_out ?SS))
  (fprintout t " ** Black out at " ?SS " ** " crlf))

(defrule black_out5
 (declare (salience 820))
 (blk_out1 ?SS ?ID)
 (not(checkcount 0))
 (not(blk_out ?SS))
 (or ?f1<-(other_line ?ID2 ?cn)
(and ?f1<-(other_lines ?ID2 ?cn)
 (dt_bs_sc ?ss&~?SS ?con ? ?ID2)
 (dt_br ?ss $? ?con $?)
 (not(dt_br ?SS ?brkr $? ?con $? C))))
 (checkcount ?n)
?f<- (chknext ?m)
=>
  (retract ?f1 ?f)
  (if (= ?n ?m)
   then
  (assert(blk_out ?SS))
  (fprintout t " Black out at " ?SS crlf)
   else(assert(chknext =(+?m 1)))))

(defrule clean_others_on_black_out
(declare (salience 818))
     (blk_out ?SS)
  ?f<-(alarm $? ?SS $?)

=>
    (retract ?f))


; ****** Rules Defineng Voltage High and Low case ********

(defrule volt_var_alma
   (declare (salience 500))
  (alarm ?date ?time ?SS bus ?ID1 KV ?nom_KV al ?val1)
   (alarm ?dat ?tim ?SS bus ?ID2&~?ID1 KV ?nom_KV al ?val2)
   (not(volt_hi_in_sub ?SS))
   (not(volt_lo_in_sub ?SS))

=>  (bind ?t (/ (- ?val1 ?val2) 100))
    (if (< ?t 0)
    then
    (bind ?t1 (* ?t -1))
```

```
    else
    (bind ?t1 ?t))
    (if (<= ?t1 1)
    then
    (bind ?val (/(+ ?val1 ?val2) 2))
    (if(> ?val ?nom_KV)
    then
    (fprintout t ?dat ?tim " Voltage is high at sub "
              ?SS " value = " ?val crlf)
    (assert(volt_hi_in_sub ?SS))
    else
    (fprintout t ?dat ?tim " Voltage is low at sub "
              ?SS " value = " ?val crlf)
    (assert(volt_lo_in_sub ?SS)))
    else
    (fprintout t ?dat ?tim " KV alarm with BTD " ?SS
           " values = " ?val1 " & " ?val2 crlf)))

(defrule volt_var_alma1
   (declare (salience 499))
  (alarm ?dat ?tim ?SS ?t&~load ?ID1 KV ?nom_KV al ?val)
   (dt_bs_sc ?SS ? ? ?ID&~?ID1)
   (not(alarm1 ? ? ?SS ?t1&~load ?ID KV ?nom_KV al ?))
   (not(volt_hi_in_sub ?SS))
   (not(volt_lo_in_sub ?SS))
=>
 (assert(volt_hi_in_sub ?SS))
 (fprintout t ?dat ?tim " KV alarm in SS " ?SS " but not at")
 (fprintout t " all equip.'s value = " ?val crlf)
 (fprintout t " suggesting that some telemetry are not
 operational in SS "  ?SS crlf))

 (defrule clean_vlt
       (declare (salience 480))
    ?a <- (alarm ?date ?time  ?SS ? ?equip_ID KV ? al ?)
          (or(volt_hi_in_sub  ?SS)
           (volt_lo_in_sub  ?SS))
=>
       (retract ?a))
;      (fprintout t ?date ?time " hi KV in sub "  ?SS crlf))

; Defining Rules to Display alarms for which no conclusion was

; reached by the IAP

(defrule display
   (declare (salience 200))
  ?a <- (alarm $?message)
=>
  (retract ?a)
  (fprintout t "alarm " $?message crlf))
```

REFERENCES

[1] B.F. Wollenberg, "Feasibility study for an energy management system intelligent alarm processor", IEEE Trans., vol. PWRS-1, No. 2, May 1986.

[2] Expert systems in Electric Power Systems - A bibliographical survey. Z.Z. Zhang, G.S. Hope & O.P. Malik, IEEE, Power Systems Nov. 1989, vol. 4.

[3] R. Fujiwara, T. Sakaguchi, Y. Kohno & H. Suzuki, "An intelligent load flow engine for power system planning", IEEE Trans. on P.S., vol. PWRS-1, No. 3 August 1986.

[4] R.P. Schulte, G.B. Sheble', S.S. Larsen, J.N. Wrubel, "Artificial intelligence solutions to power operating problems", IEEE Trans., vol. PWRS-2, No. 4, November 1987.

[5] H. Amelink, A.M. Forte, R.P. Guberman, "Dispatcher alarm and message processing", IEEE Trans., vol. PWRS-1, No. 3, August 1986.

[6] R. Davis, B. Buchanan and E. Shortlifte, "Production rules as representation for a knowledge-based consultation program", Artificial Intelligence, vol. 8, 1977.

[7] C.C. Liu, K. Tomsovic, "An Expert system assisting decision-making of reactive power/voltage control", IEEE Trans. on Power Systems, vol. PWRS-1, No. 3, August 1986.

[8] J. McDermott, "R1: A rule-based configure of computer systems", Artificial Intelligence, vol. 19, 1982.

[9] W.B. Gevarter, An Overview of Expert Systems, NBSIR 82-25-05, National Bureau of Standards, Washington, D.C., May 1982.

[10] J.H. Taylor, D.K. Fredrick, and J.R. Jame, "An expert system scenario for computer-aided control engineering", Paper WA5-12, American Control Conference, San Diego, CA, June 1984.

[11] T. Taylor, J. Wall, D. Lubkeman, "Knowledge-based expert systems for power engineering problem", Electric Power Research Center, School of Eng., North Carolina State University.

[12] D.J. Sobajic and Y.H. Pao, "An Artificial Intelligence System for Power System Contingency Screening, IEEE PAS, vol. 3, No. 2, May 1988.

[13] "Expert Systems for On-Line Security Assessment: A Preliminary Design", IEEE Trans. on P.S., vol. 3, No. 2, May 1988.

[14] C.C.Lin, S.J.Lee, S.S Venkata, "An expert system operational aid for restoration and loss reduction of distribution systems", IEEE Trans. on Power Systems, vol. 3, No.2, May 1988.

[15] B.F. Wollenberg, T, Sataguchi, "Artificial intelligence in power system operation", Invited Paper, Proc. of the IEEE, vol. 75, No. 12, December 1987.

[16] D.W. Rolston, "Artificial Intelligence and Expert Systems Development", ESL, Inc. a subsidiary of TRW, Inc. Sunnyvale, California.

[17] Carl Town Send, "Introduction to Turbo Prolog"

[18] J.R. Stefik, et,al., "The organization of Expert Systems, A tutorial". Artificial Intelligence, vol. 18, 1982.

[19] H.A. Hamli, A Rule-Based Expert System Advisor, 1985, Thesis Project, Melwauki State University.

[20] CLIPS Advanced Programming Guide, Joseph C. Giarratano, Johnson Space Center, Oct. 1987.

[21] D.G. Franz, Energy Control Center Data Acquisition & Communications Sub-systems, IEEE Tut. 1977.

[22] M.D. Anderson, Univ. of Missouri-Roll, "Power Systems Control and Protection, Implementation of the SCADA Control Center".

[23] K. Nielsen, "System operation challenges", IEEE Trans. on Power System vol. 3, No. 1, Feb. 1988.

[24] M.M. Adibi, etal., "System operation challenges", IEEE, vol. 3, No. 1, February 1988.

[25] A Bose & K.A. Clements, "Real-time modeling of Power networks", IEEE, proceedings, December 1987.