UNIVERSITY OF ALBERTA

Intelligent Maintenance Support System for Mining Trucks

by

© Andrew Wayne Ursenbach

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Science

in

Process Control

DEPARTMENT OF CHEMICAL ENGINEERING

EDMONTON, ALBERTA

FALL 1994

ISBN 0-315-95125-7

Name [illegible]

Dissertation Abstracts International is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four-digit code in the spaces provided.

CHEMICAL ENGINEERING — 0542

SUBJECT TERM — SUBJECT CODE

U·M·I

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS
Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION
General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523
Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS
Language
General 0679
Ancient 0289
Linguistics 0290
Modern 0291
Literature
General 0401
Classical 0294
Comparative 0295
Medieval 0297
Modern 0298
African 0316
American 0591
Asian 0305
Canadian (English) 0352
Canadian (French) 0355
English 0593
Germanic 0311
Latin American 0312
Middle Eastern 0315
Romance 0313
Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY
Philosophy 0422
Religion
General 0318
Biblical Studies 0321
Clergy 0319
History of 0320
Philosophy of 0322
Theology 0469

SOCIAL SCIENCES
American Studies 0323
Anthropology
Archaeology 0324
Cultural 0326
Physical 0327
Business Administration
General 0310
Accounting 0272
Banking 0770
Management 0454
Marketing 0338
Canadian Studies 0385
Economics
General 0501
Agricultural 0503
Commerce-Business 0505
Finance 0508
History 0509
Labor 0510
Theory 0511
Folklore 0358
Geography 0366
Gerontology 0351
History
General 0578
Ancient 0579
Medieval 0581
Modern 0582
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0333
United States 0337
History of Science 0585
Law 0398
Political Science
General 0615
International Law and Relations 0616
Public Administration 0617
Recreation 0814
Social Work 0452
Sociology
General 0626
Criminology and Penology 0627
Demography 0938
Ethnic and Racial Studies 0631
Individual and Family Studies 0628
Industrial and Labor Relations 0629
Public and Social Welfare 0630
Social Structure and Development 0700
Theory and Methods 0344
Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES
Agriculture
General 0473
Agronomy 0285
Animal Culture and Nutrition 0475
Animal Pathology 0476
Food Science and Technology 0359
Forestry and Wildlife 0478
Plant Culture 0479
Plant Pathology 0480
Plant Physiology 0817
Range Management 0777
Wood Technology 0746
Biology
General 0306
Anatomy 0287
Biostatistics 0308
Botany 0309
Cell 0379
Ecology 0329
Entomology 0353
Genetics 0369
Limnology 0793
Microbiology 0410
Molecular 0307
Neuroscience 0317
Oceanography 0416
Physiology 0433
Radiation 0821
Veterinary Science 0778
Zoology 0472
Biophysics
General 0786
Medical 0760

EARTH SCIENCES
Biogeochemistry 0425
Geochemistry 0996
Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES
Environmental Sciences 0768
Health Sciences
General 0566
Audiology 0300
Chemotherapy 0992
Dentistry 0567
Education 0350
Hospital Management 0769
Human Development 0758
Immunology 0982
Medicine and Surgery 0564
Mental Health 0347
Nursing 0569
Nutrition 0570
Obstetrics and Gynecology 0380
Occupational Health and Therapy 0354
Ophthalmology 0381
Pathology 0571
Pharmacology 0419
Pharmacy 0572
Physical Therapy 0382
Public Health 0573
Radiology 0574
Recreation 0575
Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES
Pure Sciences
Chemistry
General 0485
Agricultural 0749
Analytical 0486
Biochemistry 0487
Inorganic 0488
Nuclear 0738
Organic 0490
Pharmaceutical 0491
Physical 0494
Polymer 0495
Radiation 0754
Mathematics 0405
Physics
General 0605
Acoustics 0986
Astronomy and Astrophysics 0606
Atmospheric Science 0608
Atomic 0748
Electronics and Electricity 0607
Elementary Particles and High Energy 0798
Fluid and Plasma 0759
Molecular 0609
Nuclear 0610
Optics 0752
Radiation 0756
Solid State 0611
Statistics 0463

Applied Sciences
Applied Mechanics 0346
Computer Science 0984
Engineering
General 0537
Aerospace 0538
Agricultural 0539
Automotive 0540
Biomedical 0541
Chemical 0542
Civil 0543
Electronics and Electrical 0544
Heat and Thermodynamics 0348
Hydraulic 0545
Industrial 0546
Marine 0547
Materials Science 0794
Mechanical 0548
Metallurgy 0743
Mining 0551
Nuclear 0552
Packaging 0549
Petroleum 0765
Sanitary and Municipal 0554
System Science 0790
Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY
General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: Andrew Wayne Ursenbach

TITLE OF THESIS: Intelligent Maintenance Support System for Mining Trucks

DEGREE: Master of Science

YEAR THIS DEGREE GRANTED: 1994

12440 Cannes Road S.W.
Calgary, Alberta
Canada T2W OH5

Sept 15, 1994
DATE

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Intelligent Maintenance Support System for Mining Trucks** submitted by **Andrew Wayne Ursenbach** in partial fulfillment of the requirements for the degree of **Master of Science** in **Process Control.**

Dr. Ming Rao, Supervisor

Dr. Sirish Shah

Dr. Mingjian Zuo

Sept. 10, 1994
DATE

Abstract

Maintenance of heavy mining trucks at Syncrude Canada Ltd. amounts to several million dollars per year. The trucks are large with complex componentry and it is difficult for any one person to become an expert in all the maintenance procedures. Built into the trucks is a hardware monitoring system that monitors and records the performance of the vehicles. To facilitate better maintenance and reduce downtime, a project was initiated with the University of Alberta to develop an Intelligent Maintenance Support System (IMSS).

The system has three functions. The first function is condition monitoring. The condition monitoring is divided into normal condition monitoring, which displays a graphical trend of vehicle component performance, and abnormal condition monitoring, which detects faults that occur within the truck. The second function is component fault diagnosis, which narrows the conclusion of the abnormal condition monitoring to a specific component. The third function is maintenance assistance, which provides maintenance information to the user. It is an integrated system whereby the three modules work in conjunction with one another, each module using the results of the previous one. The abnormal condition monitoring and component fault diagnosis make use of expert systems, while the maintenance assistance employs a hypermedia package.

This thesis is a discussion of the development and the methodology of integration of the IMSS modules. Also discussed is the development of the supporting subsystems in the IMSS.

Acknowledgments

I would like to express my appreciation to a number of people for their valuable assistance in the completion of my thesis. First and foremost, I would like to thank my supervisor, Dr. Ming Rao, for his direction on the project. He has provided me with many opportunities for learning during the course of my research. I would also like to thank the members of my committee, namely the chairman Dr. Robert Hayes, and the committee members, Dr. Sirish Shah and Dr. Mingjian Zuo.

I would like to thank Dr. Julian Coward and David Lamb at Syncrude Canada Ltd. for their input and advice regarding the direction and needs of the project.

I would also like to express my appreciation to all the members of the Intelligence Engineering Laboratory for their advice and assistance. I would particularly like to thank Dr. Qun Wang and Haiming Qiu for always being available and willing to help me when necessary.

Finally I would like to thank my family for their help and advice in the writing of my thesis. I would also like to thank them for their support and encouragement throughout my years of education.

Financial support for this project was provided by the Natural Sciences and Engineering Research Council (NSERC), Syncrude Canada Ltd., and the University of Alberta.

Table of contents

List of figures

List of tables

1 Introduction

1.1 Syncrude mining operation and truck maintenance requirements

Syncrude Canada Ltd. has the largest oil sand extraction and upgrading plant in the world. Its oil sands mine is one of the largest surface mines. At Syncrude, a truck and shovel system with 170 to 240 ton mining trucks are used in the transportation of oil sand and overburden (McKee, 1989). In 1992, the truck fleet moved approximately 636,000 tonnes of material per day. The efficiency of the truck fleet makes a notable impact on the mining productivity and the plant-wide profits (Scoble et al., 1991; McKee 1990). Mining truck breakdowns affect not only the productivity in the mining section but also production schedules of the whole plant (Donald, 1989; Malhotra and List, 1989; Jonah, 1988). As well, maintenance of the truck fleet amounts to several millions of dollars per year. Therefore, it has become necessary that the downtime of mining trucks be reduced to minimum. Truck condition monitoring, fault diagnostics, and maintenance aids are effective tools in speeding up the maintenance process.

In condition monitoring, it is essential to be aware of the performance of many different components within the truck, With the fast growth of the mining truck size, though, there is an increasing number of system components that need monitoring (Sullivan, 1990). These components are becoming more complex in design and more difficult to maintain (House, 1990; Johnson, 1989). When a truck has a breakdown, it is difficult to quickly detect, diagnose and correct the problem.

Due to this complexity, many large mining trucks at Syncrude use a number of sensors to monitor their performance and to detect faults in various selected components. For example, on the diesel-electric 190 ton Titan T-2000™* trucks used at Syncrude, the Vital Signs Monitor (VSM™)** collects and stores 21 analog and 26 digital signals every 30 minutes, plus any time that an alarm is activated. An alarm is activated when a signal goes over or under specified limits, or when a proper condition is not met. This system of data collection generates a large number of data sets in a short period of operating time (Anon., 1992a). This data, in and of itself, is inconclusive, but is a valuable aid to operators and mechanics involved in the maintenance process.

To implement the maintenance operation automation of large trucks and to improve the productivity and availability of the trucks, the University of Alberta and Syncrude Canada Ltd. have initiated a collaborative research project to develop the Intelligent Maintenance Support System (IMSS) for truck condition monitoring and troubleshooting. This project has been supported by the Natural Sciences and Engineering Research Council (NSERC), Syncrude Canada, and the University of Alberta.

---

* Titan T-2000 is a trademark of Marathon LeTourneau
** Vital Signs Monitor and VSM are trademarks of Marathon LeTourneau

1.2 Objectives

There are four requirements defined for this project. First, there is presently no effective use for the data from the VSM™ in directing production and maintenance of the truck fleet. It is desired that some system be in place to monitor the data and evaluate its significance. Second, Syncrude expressed a need to develop a troubleshooting or diagnostic system to aid mechanics during the in-shop diagnostic process. Third, an interactive information system is needed. This would include such utilities as on-line manuals and records. Finally, there is an impetus to store the knowledge of experts in truck maintenance. Often, an employee will become an expert in some area of maintenance but take his knowledge, developed through experience and heuristics, with him when he leaves the employment of Syncrude. Protecting and utilizing this knowledge will provide assistance to other employees as well as being an effective tool in bringing new employees "up to speed" more quickly. In a nutshell, it is desired to create a system that will make the work of the employees more efficient and cost effective.

Considering the stated needs, it was decided that a computerized, intelligent system should be developed. This system would assist the users in truck condition monitoring, fault diagnosis and maintenance. The users would include operators, mechanics, engineers, and managers. The system would be an integrated system with the different functions interacting with and assisting each other. As an intelligent system, it will act on the knowledge and experience of experts in the field. It will provide all the necessary maintenance support functions at the users' fingertips.

The data from the VSM™ was necessary to the system for providing interpretive information about the condition of the truck and assisting in diagnosing specific faults.

After diagnosis, the system should provide useful maintenance information. The system should be simple and easy to understand, so that it was not, once completed, left to collect dust in a remote corner of the shop floor (Derbort et al., 1991).

The objectives of the research project, therefore, focus mainly on

- developing a new methodology for intelligent maintenance support,
- developing the applied software, Intelligent Maintenance Support System for Truck Condition Monitoring,
- setting up applied, intelligent system analysis modules for monitoring, troubleshooting and maintenance assistance,
- processing the large quantities of historical data, and
- developing a user-friendly interface using graphics and plain English.

1.3 IMSS software and benefits

The development of the Intelligent Maintenance Support System (IMSS) has meant the development of software for an integrated monitoring, troubleshooting and maintenance assistance environment. The software is still in the prototype stage, but it fulfills the stated requirements, and will hopefully result in decreased downtime for the mining equipment and net savings for the company. IMSS runs under a Microsoft Windows™* platform on an IBM compatible computer. The system employs artificial intelligence reasoning utilities, hypermedia tools, and numerical computation techniques. It also makes use of expert knowledge about truck maintenance and production.

* Microsoft Windows is a trademark of Microsoft Corporation

The condition monitoring is beneficial as it will help the user understand alarms and their significance and also provide the opportunity to observe performance history. This will enable the user to catch problems, in some cases, before they happen, which will lead to predictive maintenance (Kumar et al., 1988). With the generalization of data formats in IMSS, this system can be modified to future hardware systems that will replace the VSM™ in future trucks.

The fault diagnosis will be of a benefit to both new and experienced employees. The benefit will be through the intelligent troubleshooting using rules developed from the knowledge and heuristics of experts. The diagnosis also uses case histories and their incident reports.

The maintenance assistance will help novices understand the truck structure and components and speed up the referencing process for more experienced users. The hypermedia provides easy browsing of on-line manuals and other hypermedia information systems. Incident reports, another facet of the maintenance assistance, keep a record of symptoms with their causes and remedies for future reference and emergency situations.

1.4 Significance of the project

IMSS is an important development. Expert systems in the field of mining truck maintenance is a new area and a recent application. In this project, a unique maintenance support methodology has been developed using integrated distributed intelligent system technology. The methodology integrates monitoring, fault diagnosis, and maintenance assistance. Of particular interest is the combination and integration of expert systems with both forward and backward chaining inference engines. Each of these forms of reasoning has its respective strength. The forward chaining inference is useful in condition monitoring, while backward chaining inference lends itself well to fault diagnosis. In this system, the condition monitoring function acts to reduce the search space for the fault diagnosis inference engine. This is an effective technique, where one artificial intelligence tool is used to narrow the solution for another (Senjen et al., 1993).

The hypermedia and data handling facilities in the system make the data and information from the limited accessibility of the supplied VSM™ software and maintenance manuals available to a variety of users, and presents them in a clear and understandable fashion. Since its development, hypermedia has often been used for educational purposes but it is also an effective tool in engineering applications.

Many previous equipment maintenance systems have been DOS based. Bringing these tools together under a Microsoft Windows™ environment is consistent both with the trends of the personal computer industry and with the requirements of Syncrude.

1.5 Topics covered in this thesis

This thesis is a discussion of:

- intelligent systems (Chapter 2),
- the methodology of IMSS (Chapter 3),
- the integration of the inference engines (Chapters 3, 5, 6),
- the structure of the condition monitoring and fault checking knowledge bases (Chapters 5, 6),

- the data handling and interpretation (Chapters 4, 5),
- the use and interface of the hypermedia system with IMSS (Chapter 7),
- the development of the user interface using Microsoft Windows™ (Chapters 5, 6, 7), and
- conclusions (Chapter 8).

1.6 Publications

During the course of my involvement with the project, one paper has been accepted by an international journal (Ursenbach et al., 1994a), and another has been submitted to a Canadian journal (Ursenbach et al., 1994b). In addition, one paper was presented at a conference (Zhu et al., 1993), and one has been accepted for a conference in the fall of 1994 (Ursenbach et al., 1994c).

2

Literature review

2.1 History of condition monitoring and fault diagnosis

A number of condition monitoring schemes and fault diagnostic activities have been utilized in the past for truck maintenance. Techniques such as vibration monitoring and oil debris monitoring have helped in quantifying some conditions that were previously observed and interpreted by operators, for example unusual sounds or shaking in the equipment. More recently, many systems have hardware in place to monitor specific quantities such as pressure and temperature. With the mechanical monitoring devices in place more accurate information may be derived, but as systems get more complex the interpretation of this information becomes more difficult, even impossible for the human expert (Walter, 1990; Milne, 1987). Computers can be used in the interpretation of these data, using numerical techniques (Marciano et al., 1990; Kar and Topuz, 1989) in some cases, or, more recently, expert systems (Granholm and Kumar, 1990). The expert system in condition monitoring, a relatively new concept, is an effective tool in interpreting large amounts of data. In past years the cost of using computers with sufficient capacity and speed to handle large knowledge bases has been prohibitive. More recently, though, desktop computers have evolved to where they may perform effectively in these capacities. It is now deemed cost effective to produce an expert system to assist in maintenance and diagnostic activities (Billington, 1990; Jiao et al., 1993).

2.2 Review of intelligent systems

Engineering applications such as maintenance and troubleshooting require more than just numerical analyses. A large amount of symbolic processing is also needed. This is due to the difficulty in breaking down a maintenance problem to a mathematical algorithm. The solution of a real problem requires other sources of knowledge, such as experience and heuristics of experts. An expert has been described as:

> a person who, because of training and experience, is able to do things the rest of us cannot; experts are not only proficient but also smooth and efficient in the action they take. Experts know a great many things, and have tricks and caveats for applying what they know to problems and tasks; they are also good at plowing through irrelevant information in order to get at basic issues, and they are good at recognizing problems they face as instances of types with which they are familiar. Underlying the behavior of experts is the body of operative knowledge we have termed expertise (Oomen et al., 1987).

One branch of artificial intelligence is the development of expert systems. The purpose of an expert system is to emulate this description of an expert (Waterbury, 1988).

Conventional programs are based on numerical algorithms while expert systems, as stated, depend on heuristics and experience of experts. Expert systems are capable of symbol processing, inferencing, and explaining. This means the processing may be carried out using language and symbols that the user can easily understand (Hu, 1989). Another approach to differentiating between an expert system and a conventional software program is that an expert system is declarative rather than imperative. Declarative means a program solves a problem without specifying a procedure. Such would be the case where the flow of the program follows the line of reasoning in a set of rules input by the developer. Imperative, on the other hand, means that the program follows a series of precise commands to reach a solution (Karwatzki, 1987).

Representation of the knowledge of experts lends itself well to production rules in the form "IF premise 'A', THEN conclusion 'B'" (Schildt, 1987; Alty and Coombs, 1984). Using the production rules, knowledge can be represented by an English language sentence making the reasoning process easy to understand for the user. An expert system performs its decision making using these rules and available data.

Another form of knowledge representation is the frame structure (Alty and Coombs, 1984). A frame is a data structure with fields, or slots, describing the properties of the component or rule represented by the frame. Slots may contain information such as a variable name, value, or type, as well as any other pertinent information relevant to the object described in the frame. Slots may also contain information about the relationship of a frame to other frames. The experts system reasons using information within these frames.

As previously stated, there is both numerical computation and symbolic reasoning occurring in the solution of a real engineering problem (Figure 2.1.)

The different numerical computation components and reasoning systems need to be brought together by a coordinating program known as a coupling system (Rao et al., 1993) (Figure 2.2). A coupling system makes available the results of one package to another, interpreting the results as necessary for common use.

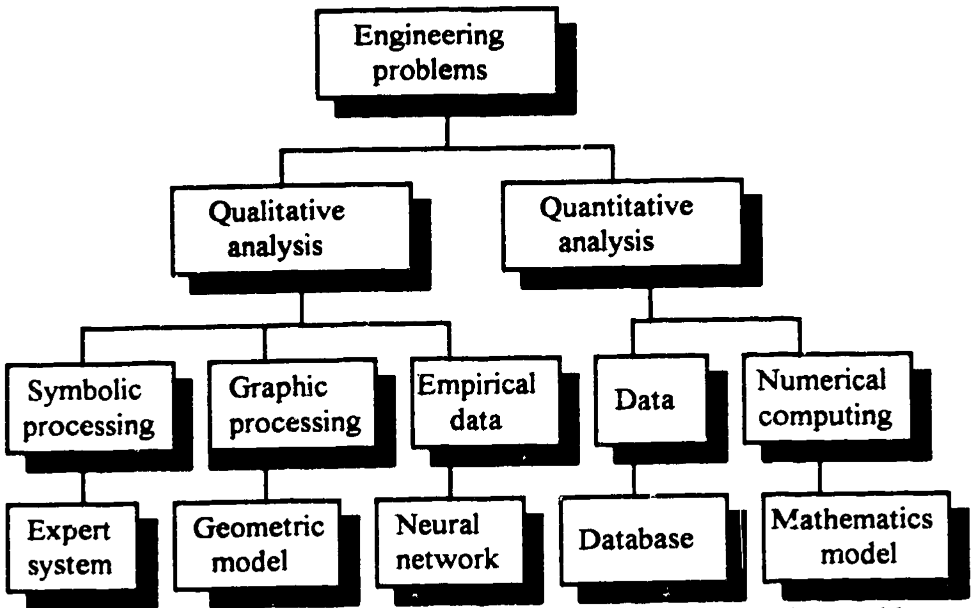

Figure 2.1: Qualitative and quantitative analyses in engineering problems (Rao et al., 1993)

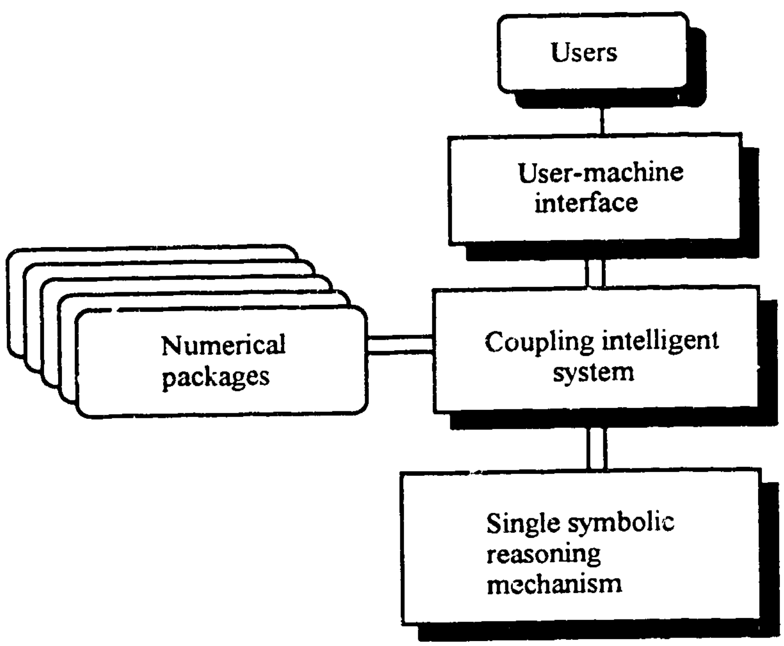

Figure 2.2: Structure of a coupling system (Rao et al., 1993)

Often, an engineering problem will use more than one type of artificial intelligence tool, as well as several types of numerical packages (Landauer, 1992). These packages and tools need to be interfaced to each other and to the user. This is accomplished through an Integrated Distributed Intelligent System (IDIS). Rao (1993) has described IDIS as follows:

> Integrated Distributed Intelligent System is a large-scale knowledge integration environment, which consists of several symbolic reasoning systems, numerical computation packages, neural networks, database management subsystem, computer graphics programs, an intelligent multimedia interface and a meta-system. The integrated software environment allows the running of programs written in different languages and the communication among the programs as well as the exchange of data between programs and database. These isolated intelligent systems, numerical packages and models are under the control of a supervising intelligent system, namely, the meta-system. The meta-system manages the selection, coordination, operation and communication of these programs (Rao et al., 1993).

2.3 Survey of intelligent monitoring and diagnosis systems

Having reviewed the purpose and structure of intelligent systems, a survey of literature regarding the use of expert systems in monitoring and diagnostic applications follows. Some of the systems listed are specific to the mining industry, while the majority are for use in other mechanical systems and with other equipment. Aspects of these systems were considered when developing IMSS.

2.3.1 Dragline Expert System (DES) (Bandopadhyay et al, 1990)

The Dragline Expert System (DES), for a walking dragline was developed in Prolog and implemented in VAX 11/750™* computers. It assists in repairs of the dragline's base component system, compressed air system, lubrication system, hoist, drag, swing and propel machinery systems, and other mechanical components. The expert system accepts a description of the problem, initiates a dialogue to acquire dragline conditions, deduces additional information from its own knowledge base and proposes a solution.

The problem solving approach used in this system is a three-level network. In this case, the three levels of knowledge are information, hypotheses, and solution levels. The information level is the level including observations, tests, desires, specifications, data, signs and symptoms. The hypothesis level includes such items as physical states, goals, and deductions. The solution level describes remedies, repair procedures, plans, design, predictions, and treatment.

The developers of the system feel there is signifance in the reasoning methods used. The methods considered included rule-based and abduction-based methods, of which it was felt that the superior method was the abduction-based, which is the method used in DES. The rule-based and abduction-based methods are, after close examination, the same as the forward chaining and backward chaining reasoning strategies, respectively. Both of these are employed for their different strengths in IMSS. Forward and backward chaining reasoning are explained in more detail in Chapters 5 and 6.

Improvements which are being considered include: 1) probability or certainty factors

* VAX 11/750 is a trademark of Digital Equipment Corporation

(Shortliffe, 1976), 2) troubleshooting of additional systems, 3) use of computer graphics, 4) adequate explanation facilities, and 5) use of sensor technology.

2.3.2 Diagnostic Maintenance Expert System (DIMES) (Mitchell, 1991)

The US Bureau of Mines has developed a sensor-based diagnostic maintenance expert system (DIMES) for the hydraulic system of a continuous mining machine. It is a frame-based system developed using a Microsoft Windows™ based expert system shell. The frames have three levels. The first represents the entire hydraulic system, the second represents the components in the hydraulic system, and the third stores the information that works with the diagnostic rules to determine machine problems. Hardware sensors have been installed in the machinery with a C language interface between the hardware system and the software diagnostics.

A diagnostic session is triggered by a sensor going out of range, a machine failure, or by a user request. At the beginning of a diagnostic session, the machine sensor data is transferred to the expert system for analysis by the diagnostic rules. DIMES will then inform the user to the condition of the machinery and provide the option to request remedy information.

2.3.3 Ride Expert (RE) (Rawicz and Jiang, 1992)

This is a PC-based expert system for diagnosing causes or at least reducing the cause search space for rough ride problems in heavy trucks. RE implements two levels of reasoning: shallow, using information acquired from operator observations, and deep or causal, which reasons using vibration information from the Multi-probe Vibration Analysis System (MVAS) (Rawicz et al., 1990).

Symptoms are first entered by the user, after which RE uses IF-THEN production rules to determine an initial diagnostic hypothesis. This is the shallow or heuristic level of reasoning. From this hypothesis, RE invokes the deep or causal level which gathers raw and pre-analyzed vibration data from MVAS and narrows or confirms the hypothesis of the shallow level. Again, IF-THEN production rules are used to organize the knowledge in the deep level. As the causal reasoning functions on a more detailed level, the complexity of the reasoning becomes massive in both a spatial and a computational sense. For this reason, the shallow reasoning is carried out first to reduce the search space of the more precise causal reasoning. Because the causal rule base has potential for growing to an excessive size, the developers of RE are considering an application of fuzzy logic. The fuzzy logic approach would be more efficient, but would result in more ambiguity in the final diagnosis.

2.3.4 Diagnostic and decision support systems for the trucking industry (ShopCoach™) (Wing and Uttamsingh, 1990)

ShopCoach™* is a tool for developing diagnostic decision support expert systems using model-based reasoning. The major components of this system are

- knowledge acquisition system,
- shop bay computer system to help the mechanic in diagnostics and repair,

* ShopCoach is a trademark of Synctics Corporation

- repair facility system to coordinate and record work and repair sessions, and
- component manufacturers system for information on reliability and maintainability of truck components.

Model-based reasoning is used in this application rather than rule-based methods. In model-based reasoning, the engineer works from design information to input the normal behavior of the device. Deviations from the proper input and output characteristics are determined as faults. Thus, a model-based system is based on understanding how a device works, rather than how it fails. In developing the knowledge base, all replaceable parts are input into the system along with working conditions, testing methods, connectivity with other components, and other data regarding the functioning of the component.

In its reasoning process, the ShopCoach™ prompts the user for the high level device that is suspected to be faulty, then the high level symptom associated with that component. If it is unable to draw a conclusion from that information, it instructs the user to perform certain tests to further verify the condition of the machinery. The user at any time may review, backtrack or exit the line of reasoning. A log is created of each diagnostic session for future reference.

The system interface uses hypertext to provide help functions and on-line documentation. There is also an information management system which develops reports for use by both the repair facility and component manufacture management.

2.3.5 Internal Combustion Engine Diagnostic Expert System (ICX) (Anvar and Kuhnell, 1991)

This is a knowledge-based expert system known as ICX for fault troubleshooting of automobile internal combustion engines. It is intended for use in automotive service departments where the users will already have some expertise in the area. The expert system is viewed as having four components:

- acquisition module, for gathering knowledge from experts,
- operator interface, for the user to view the reasoning process,
- knowledge base, a series of IF-THEN production rules, and
- inference engine, for selecting rules and carrying out inferencing.

The expert system can reference and control database fields (including dBase IV™*), spreadsheet cells, statistical variables and program variables or arrays. It can be accessed or consulted from within a program or interactively by a user.

ICX performs the fault analysis in a hierarchical fashion, whereby the engine is broken down to a series of subsystems. The knowledge about the engine consists of structural and functional information about the object as well as human experience and expertise. The diagnosis is accomplished through a series of questions to the user to narrow down possibilities until a conclusion is reached. The expert system also assigns a certainty factor to that conclusion, based on both the certainty and completeness of the facts used to draw the conclusion. Further information of the fault is then retrieved from another database. In the questioning process, ICX provides help functions using both text and graphics.

---

* dBase IV is a trademark of Borland International, Inc.

2.3.6 Jet Engine Troubleshooting Assistant (JETA) (Halasz et al., 1992)

The National Research Council of Canada has developed the Jet Engine Troubleshooting Assistant (JETA) for use in the Canadian military for aircraft gas turbine engine maintenance. The three main components of JETA are:

- the knowledge representation,
- the reasoning strategy, and
- the user interface.

JETA organizes the knowledge in a hierarchical decision tree structure that integrates the reasoning strategies of experts with the manufacturer's equipment manuals. It is a multi-level tree with each level representing a step in the reasoning process. As an example, the first level could contain a phase of operation where a fault is occurring, such as startup or acceleration. The second level would include major symptoms or "snags" that could occur in that phase of operation, such as nozzle dance or nozzle jitter during acceleration. The third level would include tests that the user needs to perform or observations that he or she needs to make to determine the cause of that error. The final level would be a component level, where the nodes contain components that should be replaced, repaired or adjusted.

Each of the nodes in the network is represented by a frame. The frames contain information such as hierarchical order, test time estimates, hypertext information, commands to the diagnostic shell and other information to aid in the diagnostic process. The developers chose a frame structure to allow for maintainability and ease in object orientation.

The reasoning module traverses the network and performs actions indicated in the frames, actions including processing directions for the user interface module, acquisition of new parameters, displaying advice, and evaluating rules. Various other modules in the functional architecture include an explanation module, a node activator, and a fault prediction module. The fault prediction module, which is still being developed, will compare current data with historical data. The user may view the reasoning process all at once to gain an overall understanding of the strategy. The operator has the option of following the questioning process through, drawing his own conclusions, or abandoning the line of reasoning and moving to a new approach.

The system, currently operating on Sun Workstations^{TM*}, uses a multi-window user interface that partitions the screen into a number of windows. The windows have different designations for displaying such features as diagrams, advice, commands, and input prompts.

2.3.7 An AI tool and its applications to diagnosis problems (IDMON^{TM}) (Narazaki et al., 1990)

An expert system shell called IDMON^{TM**} has been developed for a personal computer using the C language. Applications developed using this shell include operational guidance of the heat level control of a blast furnace, and troubleshooting a surface inspection machine.

---

* Sun Workstations is a trademark of Sun Microsystems, Inc.
** IDMON is a trademark of Electronics Research Laboratory

IDMON™ represents knowledge using three types of objects, namely hypotheses, facts, and rules. The hypotheses and facts have certainty factors associated with them. Each fact has a certainty factor used to determine the overall certainty factors of the hypothesis.

Knowledge objects are organized in two formats, those formats being a fault tree representation and a cause-symptom matrix representation. A tree structure is useful in representing knowledge in a definite and hierarchical fashion, where the verification of each intermediate hypothesis further narrows a conclusion. In the case where a number of facts need to be gathered and an analysis subsequently performed on those facts, which is often how an expert will reason, a cause-symptom matrix is a more effective representation. The expert system can use a combination of these forms for efficient knowledge representation, narrowing the problem using a tree structure, then refining it through the matrix structure. IDMON™ performs inference using a backward chaining methodology, where the inference engine verifies a hypothesis against given facts.

Optional utilities may be placed in the rules, facts, and hypotheses. These are for such purposes as providing additional explanatory information and inserting system commands. The system commands enable interfacing of the inference engine with other utilities.

In the blast furnace application, IDMON™ recommends a course of action using a heat level judgment predetermined by preprocessing and inferencing utilities. This application uses only a decision tree for representing knowledge. The surface inspection machine troubleshooting, on the other hand, uses a question and answer process with the user and refines its conclusions using both a decision tree and a cause-symptom matrix.

2.3.8 Applications of expert systems for health monitoring of machines by Trolex Ltd. (Billington, 1990)

The Trolex company has developed an expert system shell and has applied it in several machine condition monitoring applications, including helicopter vibration and air compressors in an automotive plant.

The major elements of their knowledge representation are identifiers, facts, rules, questions and default values. Identifiers are names of entities which have values, or variables. Three basic forms of expression exist in the expert system, namely numeric, symbolic, and assertion, the latter being a textual statement that can be either true or false. Uncertainty is expressed in the wording of the knowledge.

Also developed is a data acquisition and reduction module which places the information in a market spreadsheet. When performing a consultation session, an application may either access the knowledge base directly or it may access it indirectly through a software linkage. At any time the user can interrupt the process and view the history of the consultation, then resume the process.

2.4 Integration of condition monitoring, fault diagnosis, and maintenance assistance in the IMSS

IMSS is to be an integrated system embodying many of the properties of IDIS. It is desired to have a complete system that will provide easy access for the user to different maintenance tools as well as having the tools exchange information among themselves.

The IMSS tools were developed for the distribution of the following tasks: 1) normal condition monitoring for viewing of trends, 2) abnormal condition monitoring for intelligent interpretation of machine signals, 3) component fault diagnosis for assistance in diagnosing defects in the truck components, and 4) maintenance assistance tools for providing procedural and technical information. Each module functions using information from the previous one.

The main efforts of this thesis project have been to define and develop these components and the framework surrounding them. The following chapters describe the resulting intelligent system. A complete explanation of the methodology of integration of IMSS's modules follows in Chapter 3.

3

IMSS architecture

3.1 Overview of IMSS

A practical methodology for equipment troubleshooting and maintenance has been developed, as shown in Figure 3.1.

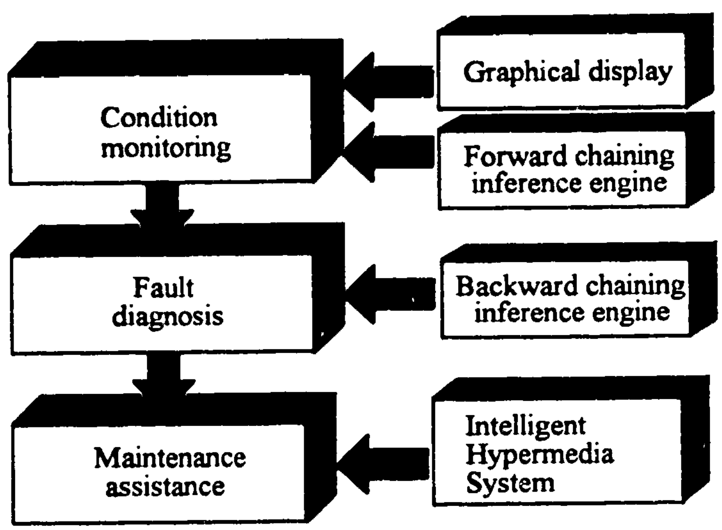

Figure 3.1: IMSS methodology

In this diagram the primary components of the methodology are the condition monitoring, fault diagnosis, and maintenance assistance. The tools employed for the functioning of these modules are a graphical trend display routine, expert systems with forward and backward chaining inference engines, and an intelligent hypermedia system. The forward chaining inference engine is used with the condition monitoring system while the backward chaining engine is used for the fault diagnosis.

The modules work in conjunction with one another, with the results of the previous stages of operations being used in subsequent stages. The following figure details the integration of this methodology in IMSS. It outlines the flow of information in the system and the maintenance process. (Figure 3.2)

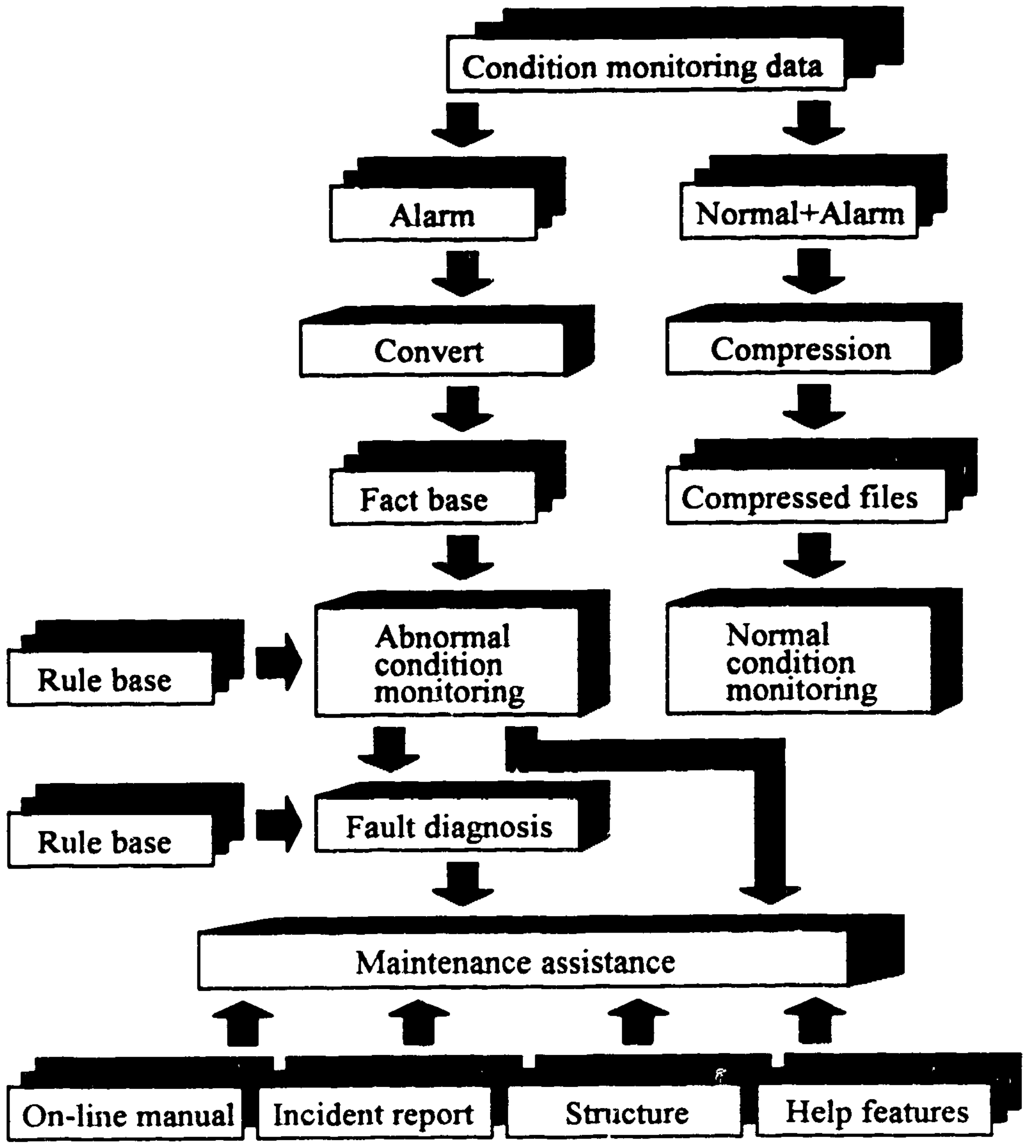

Figure 3.2: IMSS data flow and overall structure

In IMSS, three types of condition monitoring are discussed. The top level condition monitoring in Figure 3.2 is the hardware condition monitoring. This is composed of actual sensors built into the truck that monitor specific properties such as crankcase pressure or motor voltage. The other two types of condition monitoring are software based. The normal condition monitoring is a trend display routine while the abnormal condition monitoring is an interpretive decision tool. Both use data from the hardware condition monitoring.

In the prototype of IMSS, the hardware condition monitoring system used is Marathon LeTourneau's Vital Signs Monitor. The VSM™ data is first processed by the conversion or compression utilities for use by subsequent stages in the system. Under a normal situation, the normal condition monitoring module draws the curve that displays the trend of each monitored variable. When an abnormal case occurs the inference engine, using the condition monitoring rule base, provides a hypothesis of a possible problem area in which a fault has occurred or is likely to occur. The abnormal condition monitoring inference engine deals only with the data from the sensors. At this time, it is also an off-line process where the data must be downloaded to the computer.

The step following the abnormal condition monitoring uses the component fault diagnostic inference engine to further analyze components in the hypothetical failure area. The component fault diagnosis module uses a rule base developed specifically for the failure area. In providing an area of failure for the component fault diagnosis to analyze, the abnormal condition monitoring acts to reduce the search space for the fault diagnosis inference engine. In practice, diagnosing the causes of a defect is a difficult undertaking. The fault diagnostics will assist in this task by asking the operators or the mechanics for more detailed information obtained by direct observation or indirect reasoning. If the user is unable to answer the question or does not understand or would like clarification on the question, a help function is always available for additional explanation or for an outline of testing procedures. Once all the information is gathered the fault diagnostics then provides information concerning the state of a specific component.

If a component fault has been verified, IMSS can automatically open the intelligent hypermedia system (IHS) to display all information and data about removal, installation, testing, and maintenance of the faulty component. It will also display information regarding the particular fault and specific procedures involved with its remedy. This is one facet of the maintenance assistance. The maintenance assistance also provides tools for browsing the truck structure, case histories for troubleshooting, and on-line help functions.

Although there is a flow of information from one module to the next, the distributed nature of IMSS also allows for the functioning of each component independent of the others. Any one of the modules, normal and abnormal condition monitoring, fault diagnosis, or the hypermedia applications, may be accessed directly from the main IMSS menu.

3.2 System functions and configuration

A summary of the functions of IMSS detailed in Section 3.1 is as follows: (1) normal and abnormal condition monitoring, (2) fault diagnosis, (3) maintenance assistance and on-line maintenance manual display. These functions are intended for use by novices in truck maintenance, truck operators, truck maintenance mechanics, and section reliability managers at Syncrude Canada Ltd. The interface of IMSS must be easily useable by all these parties.

IMSS has therefore been developed in the Microsoft Windows™ platform. Microsoft Windows™ is rapidly becoming the standard due to its ease of use and its ability to coordinate and integrate many functions. It is also the preferred platform at Syncrude which already uses a number of 486 based personal computers with adequate memory. The main module consists of three levels of sub-menus (Figure 3.3), which the various users may operate.

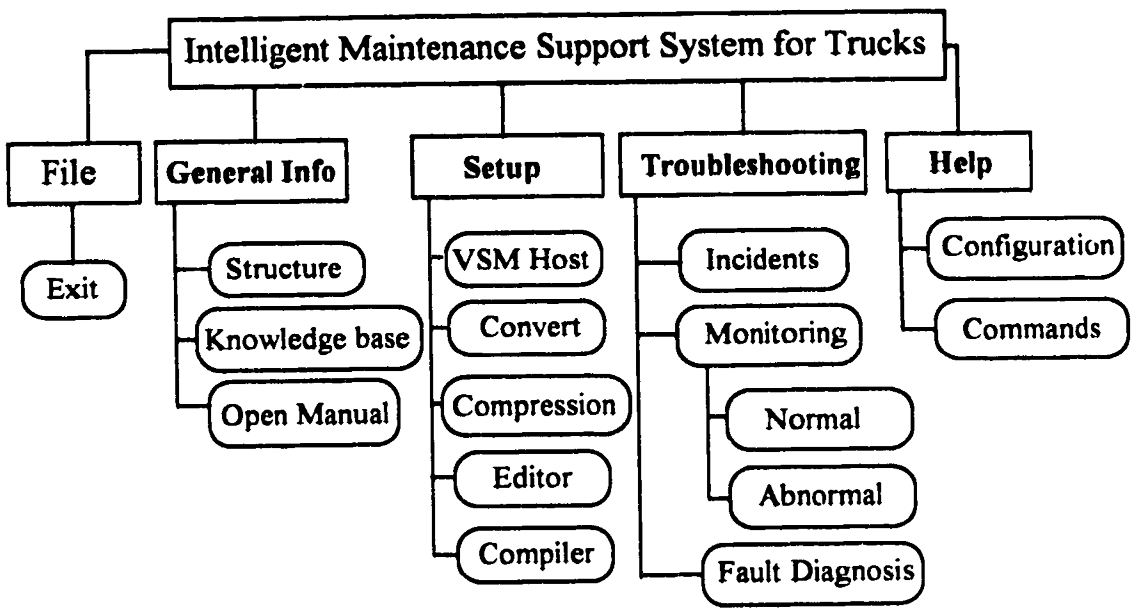

Figure 3.3: IMSS menu system

In IMSS, the modeling efforts focused on the development of the monitoring and diagnostic knowledge bases, data handling techniques, and hypermedia facilities. To date, the following sub-systems have been developed:

- knowledge base for truck condition monitoring,
- knowledge bases for diagnosis of the steering and hoist systems,
- data compression and conversion modules,
- structural diagrams,
- interactive troubleshooting module for the hydraulics using incident reports, and
- on-line maintenance manual for the hydraulic system.

The following paragraphs describe the various functional modules accessible through the menus of IMSS. The order of the descriptions follows the order in the IMSS menus.

Structure: This module describes structures, functions and location of each component of the truck. Using hypermedia, it can provide on-line information for the users with detailed textual and graphical descriptions of components which they may browse.

Knowledge base: Selecting "Knowledge base" from the main menu provides a facility to help the user to understand the structure and contents of the knowledge bases and the relationships among them. This has not yet been developed.

Open manual: This is an on-line maintenance manual for T-2400 trucks that contains general information, as well as information on operation and service of mechanical, hydraulic, electrical and electronic components.

VSM Host™: This calls a batch file which runs the DOS-based VSM Host™* software package supplied by Marathon LeTourneau. Host™ displays the data and information from the sensors and can store the sensor information in a text file usable by the data handling components of IMSS.

Convert: The "Convert" option transfers data from VSM™ format to the specified database or fact base format used by the inference engines.

Compression: This function reduces the historical data from the sensors. The data compression model can compress the number of data points to 60% or better of the

* VSM Host is a trademark of Marathon LeTourneau

original number of data points in the VSM™ files, depending on the nature of the original data and compression parameters. It also parses the data into individual files for each variable. These files are created for easy handling by the normal condition monitoring routine.

Editor: This is simply a text editor window used to build and edit ASCII knowledge base files.

Compiler: This compiles ASCII knowledge base and fact base files, transferring them to the internal form used by the inference engines.

Incident report: This hypermedia function provides case histories of previous faults. The module shows a list of fault symptoms with respective causes and remedies. Other reporting facilities are possible here including maintenance checklists and report forms.

Condition monitoring -- normal: This graphically provides the trend of each analog variable from the compressed files. Display options are available such as scaling of the graph and point and click interpretation of the data points.

Condition monitoring -- abnormal: When an abnormal or alarm situation occurs, this module provides a possible interpretation using the condition monitoring inference engine, the condition monitoring knowledge base, and the data from the VSM™. The result is a conclusion about a general area of the truck in which a fault is likely to occur. This conclusion directs the reasoning of the fault diagnosis.

Component fault diagnosis: The abnormal condition monitoring calls the fault diagnosis module by the "Link" option in the condition monitoring menu. The component fault diagnosis can also be called from the main menu, i.e. from the "Troubleshooting" menu. The fault diagnosis uses the probable defect area ascertained by the condition monitoring to identify the proper knowledge base. This knowledge base is subsequently used to further narrow the defect to a specific component. The component fault diagnosis can then access the hypermedia system to provide further information on the fault and the associated remedy.

Configuration: This will describe the configuration of IMSS using the intelligent hypermedia system. It has not yet been developed.

Commands: This on-line help gives a detailed explanation of each menu item. It provides a diagram of the menu options available, similar to the diagram shown in Figure 3.3. The user may obtain information on any component of the menu by merely clicking the mouse cursor on the respective menu item in the diagram.

4

Data processing

4.1 Purpose and functions

As stated in the introduction, a Vital Signs Monitor (VSM™) system is built into the Marathon LeTourneau T-2000™ series trucks for the purpose of condition monitoring. Tables 4.1 and 4.2, respectively, outline the 21 analog and 26 digital signals monitored by the VSM™ (Anon., 1992a).

Table 4.1: Analog signals monitored by VSM™

Number	Variables	Units	Description
A-1	COOL TP	DEG C	Engine coolant temperature
A-2	OIL PSI	PSI	Oil pressure
A-3	ENG RPM	RPM	Engine RPM
A-4	HP	HP	Horsepower
A-5	+24-MN	VDC	24 volts-main
A-6	INT TMP	DEG C	VSM™ internal temperature
A-7	FC TEMP	DEG C	Motor field converter temperature
A-8	FLD AMP	AMPS	Motor field current
A-9	AC1 TMP	DEG C	Armature converter 1 temperature

Table 4.1: Analog signals monitored by VSM™ (continued)

A-10	MT1 AMP	AMPS	Motor 1 current
A-11	MT1 RPM	RPM	Motor 1 speed
A-12	MT1 TMP	DEG C	Motor 1 temperature
A-13	MT1 VLT	VDC	Motor 1 voltage
A-14	AC2 TMP	DEG C	Armature converter 2 temperature
A-15	MT2 AMP	AMPS	Motor 2 current
A-16	MT2 RPM	RPM	Motor 2 speed
A-17	MT2 TMP	DEG C	Motor 2 temperature
A-18	MT2 VLT	VDC	Motor 2 voltage
A-19	GEN TMP	DEG C	Generator temperature
A-20	GEN VLT	VAC	Generator voltage
A-21	VR TEMP	DEG C	Voltage regulator temperature

Note: DEG C - Degrees Celsius
PSI - Pounds per Square Inch
AMPS - Amperes
VDC - Volt (direct current)
DEG F - Degrees Fahrenheit
RPM - Revolutions Per Minute
VAC - Volt (Alternating Current)

Table 4.2: Digital signals monitored by VSM™

Number	Variables	Status	Description
D-1	COOL L	LO OK	Engine coolant level low Engine coolant level OK
D-2	COOL P	LO OK	Engine coolant pressure low Engine coolant pressure OK
D-3	CCPRES	HI OK	Crankcase pressure high Crankcase pressure OK
D-4	OIL LV	LO OK	Oil level low Oil level OK
D-5	EN FNC	HI LO ME	Engine radiator fan speed command high Engine radiator fan speed command low Engine radiator fan speed command medium
D-6	ENGTSW		Unavailable
D-7	BODY	DOWN UP	Truck body down Truck body up
D-8	BRKDIF	HI OK	Brake pressure differential high Brake pressure differential OK

Table 4.2: Digital signals monitored by VSM™ (continued)

D-9	CL AIR	OK	Cooling air OK
D-10	RETDR	FAIL OK	Dynamic retarder fail Dynamic retarder OK
D-11	CABNT	CLS OPEN	Electrical cabinet door closed Electrical cabinet door open
D-12	FAN	CMD HIGH IDLE LOW MED	Fan speed command Fan speed high Fan speed idle Fan speed low Fan speed medium
D-13	FUEL	LOW OK	Fuel level low Fuel level OK
D-14	GROUND	OK	Ground fault current OK
D-15	ALTER	FAIL OK	Battery Alternator fail Battery Alternator OK
D-16	BRK	APPLD RELSD	Mechanical brakes applied Mechanical brakes released
D-17	STR PR	LO OK	Steering pressure low Steering pressure OK
D-18	VH SPD	OK	Vehicle speed OK
D-19	MT FLD	AL OK	Motor field alarm Motor field OK
D-20	MT/AC1	AL OK	Motor 1 / Armature converter 1 alarm Motor 1 / Armature converter 1 OK
D-21	MTR1	FWD REV	Motor 1 direction forward Motor direction reverse
D-22	MT/AC2	AL OK	Motor 2 / Armature converter 2 alarm Motor 2 / Armature converter 2 OK
D-23	MTR2	FWD REV	Motor 2 direction forward Motor 2 direction reverse
D-24	GENER	ALM OK	Generator alarm Generator OK
D-25	HY FNC		Unavailable
D-26	HYDRL	ALM OK	Hydraulic tank alarm Hydraulic tank OK

The VSM™ performs data dumps of all the variable values at fixed intervals of thirty minutes as well as at every alarm. This information is downloaded periodically to a PC and stored in a binary file with a [***.VSM] extension. The [***] in the file names is the name of the file originally downloaded from the VSM™ containing the information on the truck and date. For example, the file 57041492.VSM contains information on truck 57 downloaded April 14, 1992. The user can view these files in a limited fashion by the VSM Host™ program supplied by Marathon LeTourneau. The data file contains relevant and useful information. However this information in itself cannot provide any conclusions concerning the condition of the truck. The data, therefore, must be converted into a format useable by the condition monitoring and fault diagnosis modules of the IMSS for interpretation. The conversion and compression functions are the interfaces between the VSM™ data and the main working modules of IMSS.

4.2 Data exchange and conversion

The [***.VSM] files are of unknown format but the information from them can be stored in a text file, called a list file in the nomenclature of IMSS, by the Host™ program. This list file is indicated with a [***.LST] extension. This file contains a number of data dumps in the format shown in Table 4.3. Each data dump is referred to as a "snapshot".

Table 4.3: Data dump snapshot

```
1992 APR  9  21:56:02        BRAKE PRESSURE DIFF. HIGH
1992 APR  9  21:56:02        ALARM FULL DATA DUMP

ALTER OK      MTR1 FWD       FUEL L OK    BRKDIF HI     NO LMP TS
GROUND OK     MTR2 FWD       BRK RELSD    BODY DOWN     RETDR OK
ARM1 OK       ALUBE OK       HYDRL OK     CABNT CLS     GEN OK
ARM2 OK       CL AIR OK      STRPRS LO    OVERSD OK     MTRFLD OK
OIL LV OK     COOL P OK     FILTER OK

COOL TP     75 DEG C    OIL PSI     36 PSI      ENG RPM    850 RPM
HP       -4031 HP       +24V-MN   25.0 VDC      BAT EQL   12.3 VDC
INT TMP  17.2 DEG C     AC1 TMP   < 54 DEG C    MT1 TMP < 118 DEG C
```

Table 4.3: Data dump snapshot (continued)

FC TEMP	< 54	DEG C	AC2 TMP	< 54	DEG C	MT2 TMP	< 118	DEG C
FLD AMP	0	AMPS	MT1 AMP	0	AMPS	MT1 VLT	0	VDC
MT1 RPM	0	RPM	MT2 AMP	0	AMPS	MT2 VLT	0	VDC
MT2 RPM	0	RPM	GEN TMP	< 111	DEG C	GEN VLT	10	VAC
VR TEMP	< 54	DEG C	RC TEMP	< 54	DEG C	ROT CUR	-2	AMPS
HOIST P	-20	PSI	RC VOLT	0	VDC	RETR HP	0	HP

The abnormal condition monitoring and component fault diagnosis inference engines require the data to be in a specific fact base format. Convert, one component of IMSS, takes one snapshot at a time specified by the user from the list file and converts it into an ASCII fact file [***.FAC]. The fact file is in a form that the compiler can understand. The compiler module subsequently compiles the fact files or fact bases into the code structure understood by the inference engines. The proper form for a fact in the fact file is a three item sentence consisting of name, relationship and value. Each item may contain a number of words joined with underscores to make a single string. Table 4.4 shows a sample of the fact file derived from the snapshot in Table 4.3.

Table 4.4: Fact base format

fact : ALTER is OK	fact : MTR1 is REV	fact : FUEL_L is OK
fact : BRKDIF is OK	fact : NO_LMP is TS	fact : GROUND is OK
fact : MTR2 is FWD	fact : BRK is RELSD	fact : BODY is UP
fact : RETDR is OK	fact : ARM1 is OK	fact : ALUBE is OK
fact : HYDRL is OK	fact : CABNT is CLS	fact : GEN is OK
fact : ARM2 is OK	fact : CL_AIR is OK	fact : STRPRS is LO
fact : OVERSD is OK	fact : MTRFLD is OK	fact : OIL_LV is OK
fact : COOL_P is OK	fact : FILTER is OK	fact : COOL_TP = 72
fact : OIL_PSI = 44	fact : RPM = 898	fact : HP = 4031
fact : +24V-MN = 18.1	fact : BAT_EQL = 13.9	fact : INT_TMP = 17.2
fact : AC1_TMP < 54	fact : MT1_TMP < 118	fact : FC_TEMP < 54
fact : AC2_TMP < 54	fact : MT2_TMP < 118	fact : FLD_AMP = 0
fact : MT1_AMP = 0	fact : MT1_VLT = 0	fact : MT1_RPM = 0

Table 4.4: Fact base format (continued)

fact : MT2_AMP = 0	fact : MT2_VLT = 0	fact : MT2_RPM = 0
fact : GEN_TMP < 111	fact : GEN_VLT = 0	fact : VR_TEMP < 54
fact : RC_TEMP < 54	fact : ROT_CUR = -2	fact : HOIST_P = 79
fact : RC_VOLT = 0	fact : RETR_HP = 0	end

Other information in the same name-relationship-value format may be placed into the fact base for analysis by the condition monitoring engines. This information could include results from such tools as numerical analysis packages that would preprocess the VSM™ signals. An example would be "standard_deviation = 6." At present, no packages of this nature exist as Syncrude has not yet found any numerical relations that are useful. The possibility is accounted for, however, and the technology to accommodate them is in place. The section describing the compiler, Section 4.4, discusses the name-relationship-value fact format more completely.

4.3 Data compression

A large amount of the data generated by the VSM™ is redundant. This is due to the small variation of data values between successive snapshots. The purpose of the data compression module is to remove the unnecessary points. This reduces storage space and decreases computation time for any numerical packages which would process this data. The data compression module also parses the data into separate data files for each variable to increase the speed of access to the data by the normal conditioning monitoring unit.

The following diagram (Figure 4.1) describes the data flow:

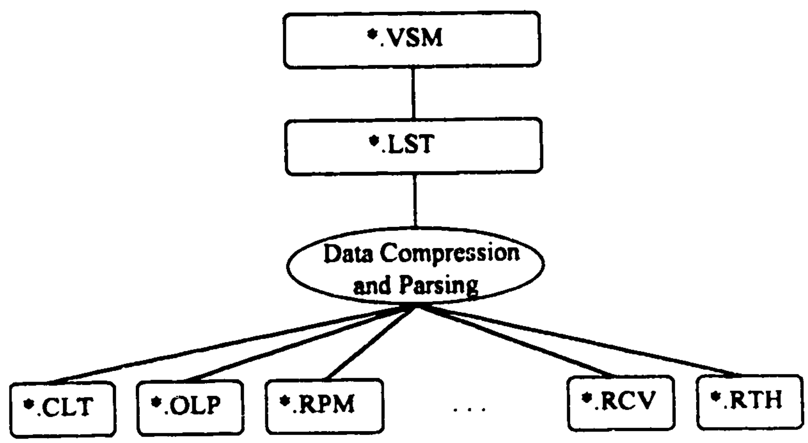

Figure 4.1: Normal data handling flow

The data is passed from the [***.VSM] format to the [***.LST] format as previously described. Each variable is then parsed from the list file, passed throught the compression routine to remove non-essential points, then stored in its own file with the extension of that file being a variable descriptor as shown in the bottom level of Figure 4.1. Table 4.5 describes the parsing of the data, giving the descriptor extension for each of the variable files.

Table 4.5: Extensions of parsed data files

COOL TP	***.CLT	FC TEMP	***.FCT	MT2 RPM	***.M2R
OIL PSI	***.OLP	AC2 TMP	***.A2T	GEN TMP	***.GNT
RPM	***.RPM	MT2 TMP	***.M2T	GEN VLT	***.GNV
HP	***.HP	FLD AMP	***.FDA	VR TMP	***.VRT
+24V-MN	***.VMN	MT1 AMP	***.M1A	RC TEMP	***.RCT
BAT EQL	***.BEQ	MT1 VLT	***.M1V	ROT CUR	***.RTC
INT TMP	***.INT	MT1 RPM	***.M1R	HOIST P	***.HSP
AC1 TMP	***.A1T	MT2 AMP	***.M2A	RC VOLT	***.RCV
MT1 TMP	***.M1T	MT2 VLT	***.M2V	RETR HP	***.RTH

IMSS stores the data in a binary file beginning with start time, end time and number of points. The variables are stored in a point structure consisting of a float variable value and an integer time value. The time for each point has been converted to an integer value in seconds using the Borland C++™* library functions (Anon., 1992b).

The data compression works using a tolerance-band established from a tolerance value presently coded into the compression routine. Any points falling within this tolerance-band are not included in the final parsed variable files. An outline of the compression procedure is as follows:

(1) The compression routine selects three consecutive points, identified as points 1, 2, and 3. A slope between points 1 and 3 is evaluated.

(2) An expected value for point 2 is established from this slope.

(3) If the difference between the actual value for point 2 and the expected value for point 2 is smaller than some tolerance, point 2 is eliminated. This is done by setting the new point 2 equal to the old point 3, and bringing in the next point as the new point 3. Control then returns to step (2).

(4) If the difference is larger than the tolerance, point 1 is saved. The new point 1 is set equal to the old point 2 and the new point 2 equal to the old point 3. The next point is brought in as the new point 3. Control is then passed back to step (2).

* Borland C++ is a trademark of Borland International, Inc.

4.4 Compiler

The purpose of the compiler module is to convert the fact [***.FAC] files into a format useable by the inference engines. The module also compiles the rule base, but a discussion of this is left to the next chapter.

As stated, all the facts are entered into the fact base using a three part sentence structure, these parts being name, relation, and value. The name is a string of up to 39 characters in length. The relation can be either an English word or phrase ("is", "are", etc.) of not more than four characters or a logical operator ("=", "<", ">", "<=", ">=", and "!=".) The value may be either a string or a numerical value describing the state or condition of the object identified in "name". For example, the first fact in Table 4.4 is "ALTER is OK". In this case, the name is "ALTER", the relation is "is", and the value is a string, "OK". All information taken directly from the VSM™ is in uppercase lettering. A numerical expression is represented by a sentence such as "INT_TMP = 17.2." The compiler reads these three item sentences, parses them, and places them in the appropriate slots in a C language data structure shown below.

Table 4.6: C language data structure of facts

```
union VALUE
{
   char progn[40];
   float fltvb;
};

struct fac
{
   char fname[40];
   char frela[5];
   int ftype;
   VALUE fvalue;
```

Table 4.6: C language data structure of facts (continued)

```
   struct fac *fnext;
   struct fac *fprv;
} fact;
```

The compiler stores the name in `fname`, the relationship in `frela`, and the value in the VALUE union. The value is stored in `fltvb` if it is a numerical value, and `progn` if it is a symbolic, or character value. The `*fnext` and `*fprv` pointers are used to create a linked list of the facts. The integer value `ftype` is not used in the facts and is assigned a value of "-1". It is in place for compatibility with the rule structure, described later in Chapter 5, to facilitate the compilation. The facts are then stored in their structures in a binary file with a [***.DB] extension, indicating a database. With the fact base in this format, the inference engine is now able to operate on this information and make a decision using the rule base.

5

Condition monitoring

5.1 Purpose and functions

There are a number of signals monitored by the VSM™. In the past, there was nothing being done with the data which has been accumulating over the years. It is desirable to start using this data to aid in predicting and understanding the faults that occur in the truck. The condition monitoring system developed to do this should be able to realize the significance of any alarms and have the ability to view trends. With this, the user can observe performance history and discover problems before they occur or before they cause significant damage or downtime. Another consideration in the development of the IMSS is that Syncrude is planning for other hardware condition monitoring systems to be installed in future trucks to replace the VSM™. Examples of alternative hardware condition monitoring systems include ROPEC™* and the Vehicle Information Monitoring System (VIMS™**). The condition monitoring component should be adaptable to other hardware condition monitoring systems built into future trucks.

* ROPEC is a trademark of Ropec Industries
** VIMS is a trademark of Caterpillar Inc.

5.2 Normal condition monitoring

As explained in the previous chapter, IMSS compresses the historical data and parses it into individual files. This data may be handled by the normal conditioning module of IMSS. When the user selects the "Normal" option from the "Condition Monitoring" menu item in the "Troubleshooting" menu, IMSS calls the routine for plotting historical data. To observe the historical data from a data dump, the common filename must first be selected. As all files are parsed from a list [***.LST] file, a file dialog box displays the available list files in the directory when the "Open" option from the "File" menu is selected. When a list file is selected, IMSS removes the "LST" extension from the filename. Thereafter, when the user selects a variable from the variable option buttons at the bottom of the screen, the monitoring program places the appropriate extension on the filename. The extensions are those indicated in Table 4.5 in the previous chapter. Figure 5.1 shows the screen when the trend of coolant temperature is selected.

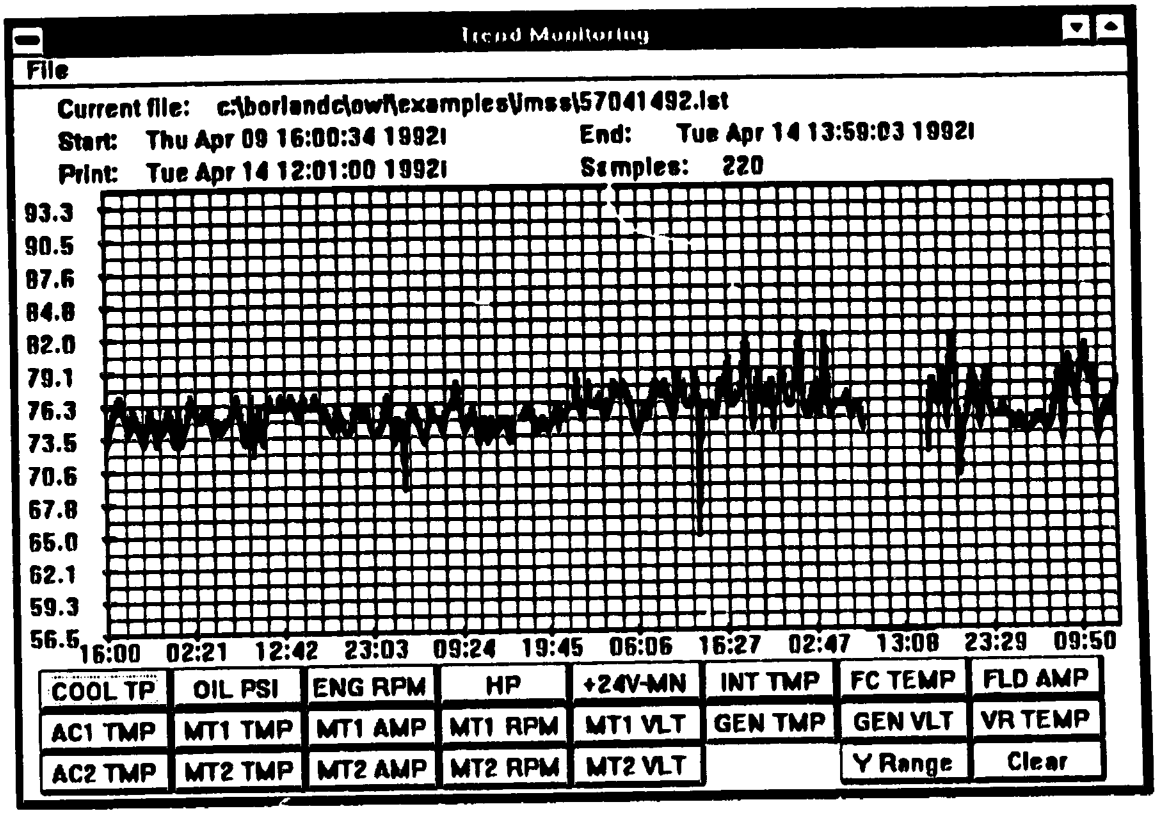

Figure 5.1: Coolant temperature with 0.1 degrees Celsius tolerance

The display shows that 220 data points are in this file. The original number recorded by the VSM™ was 265. This decrease is due to a compression tolerance of 0.1 degrees Celsius. This tolerance was coded directly into the compression routine. If the tolerance is increased to 2 degrees, the number of sample points decreases to 81, but the overall variable trend remains, as shown in Figure 5.2.

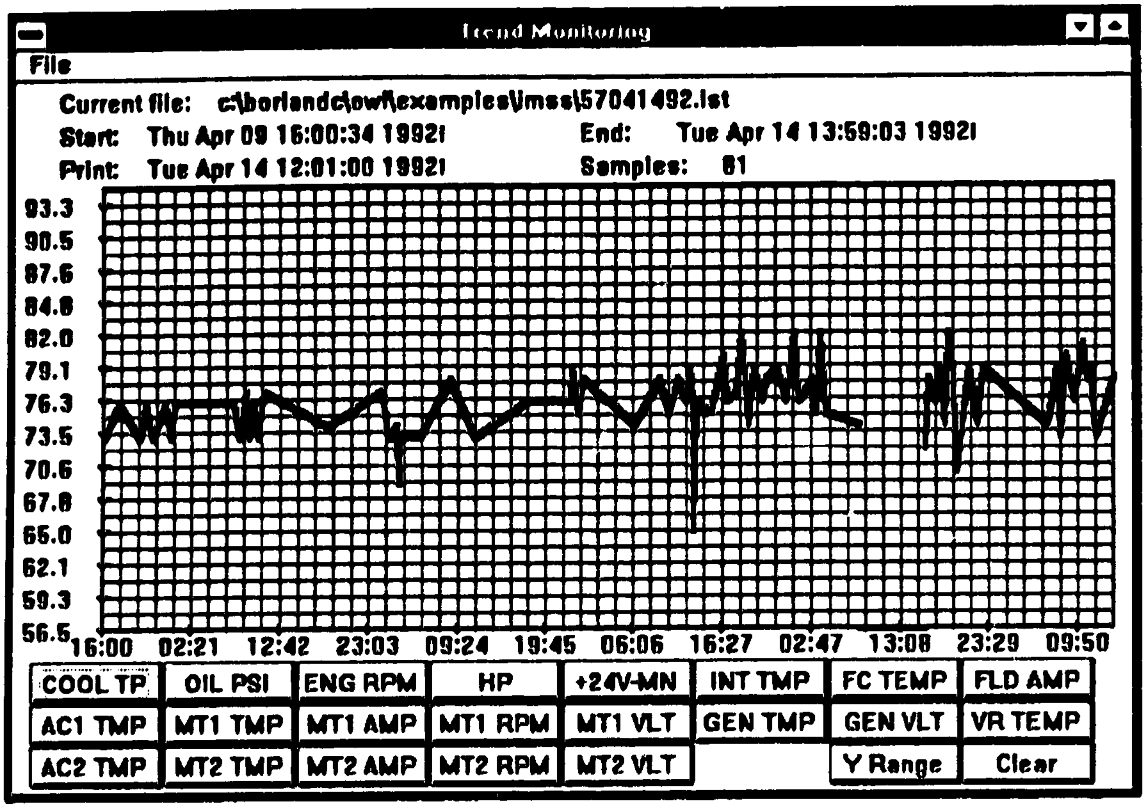

Figure 5.2: Coolant temperature with 2 degrees Celsius tolerance

The user may change the Y axis scale by selecting the "Y Range" button from the button panel. IMSS will display the old values of minimum and maximum Y presently on the grid and give the user the opportunity to change the values.

When the user moves the mouse cursor over the grid, it changes from an arrow cursor to a cross-hair cursor. Clicking this cross-hair cursor on any part of the graph will display the date, time, and value for that point. Figure 5.3 shows the screen displaying the date and time. As well, the scale of the Y-axis has been changed from Figure 5.2.

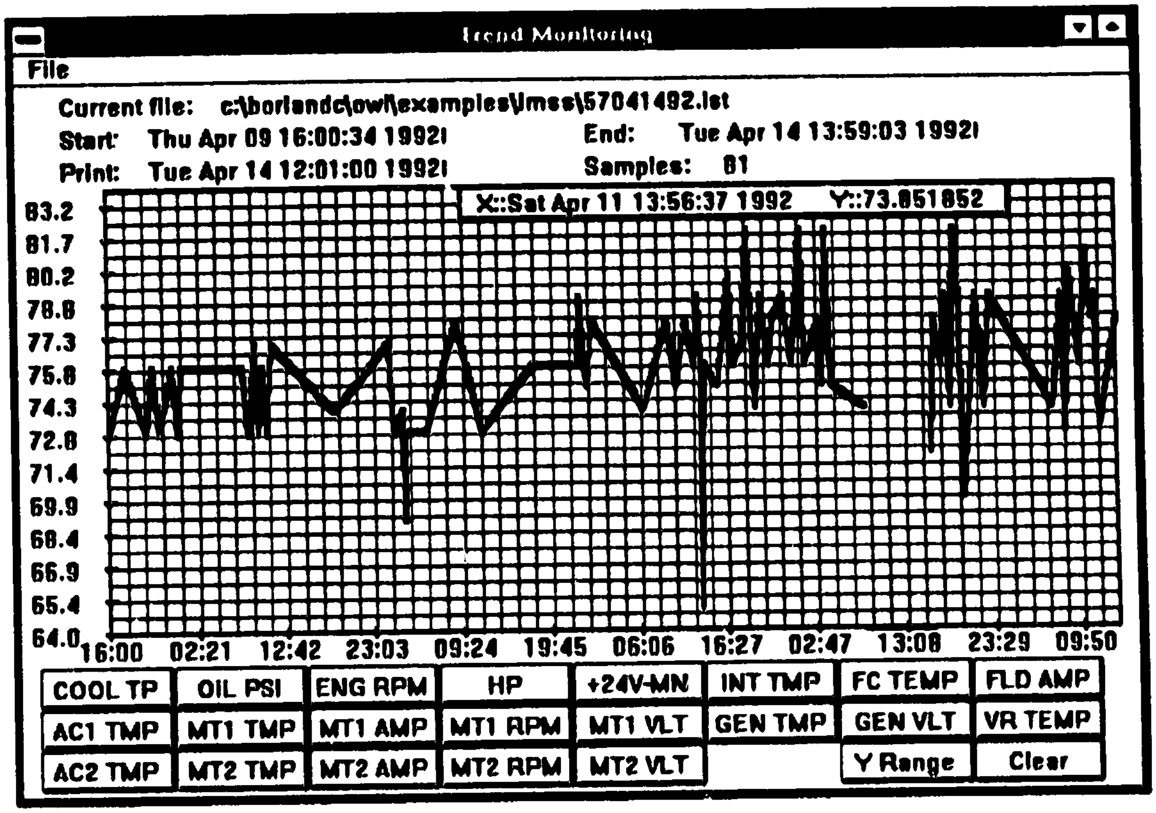

Figure 5.3: Coolant temperature with date and time display and changed Y axis

There is a periodic break in the graph and a red line at either the top or the bottom of the graph appears for the length of that break. In this case, the value was either above or below sensor range, as indicated by a greater than or less than sign in the original list file. If the user clicks the cursor on this red line, a window appears explaining this fact (Figure 5.4).

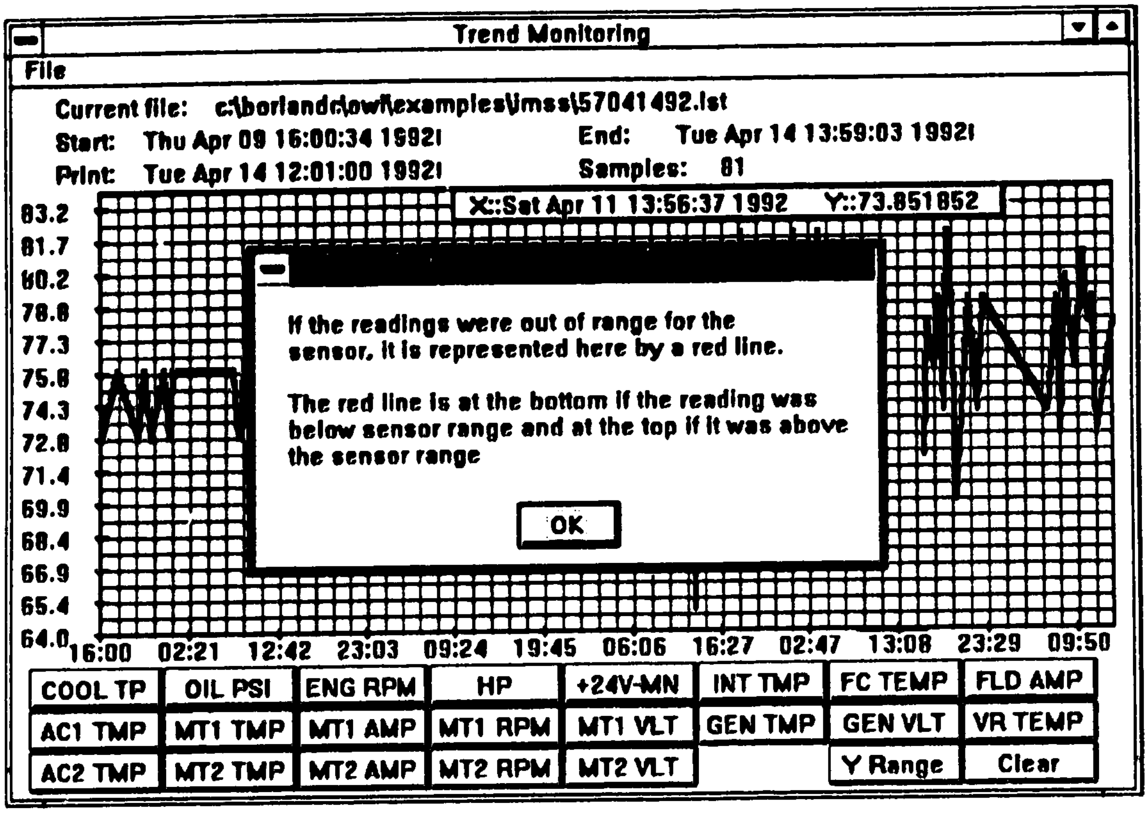

Figure 5.4: Explanation of out-of-range red line

The normal condition monitoring provides the user with the ability to track the performance of a desired variable or truck component over the history of the file. This is an effective tool in understanding the significance of alarms and will eventually aid in performing the ideal "just in time" maintenance.

5.3 Abnormal condition monitoring

As explained in Chapter 4, every time an alarm condition occurs, the VSM^{TM} performs a data dump of all the signals from the monitored components. Each data dump may be converted into the fact base format as described in the convert process. Numerical analysis packages that work on the data can place their results in the fact base. The abnormal condition monitoring engine uses this fact base to draw conclusions regarding the condition of the machinery.

As stated in the Section 3.1, the overview of IMSS, the abnormal condition monitoring and the component fault diagnosis modules are expert systems. An expert system is a computer program with the ability to receive data, reason and draw conclusions about that data. Expert systems work using a set of rules derived from the experience of domain experts. The reasoning component of an expert system is the inference engine. The abnormal condition monitoring inference engine is of the forward chaining type. A forward chaining inference engine is one that receives data or facts and infers a conclusion based on those facts. In terms of a fault tree, the leaf nodes contain facts or data and the inference engine follows the fault tree to a correct conclusion in the root node (Figure 5.5).

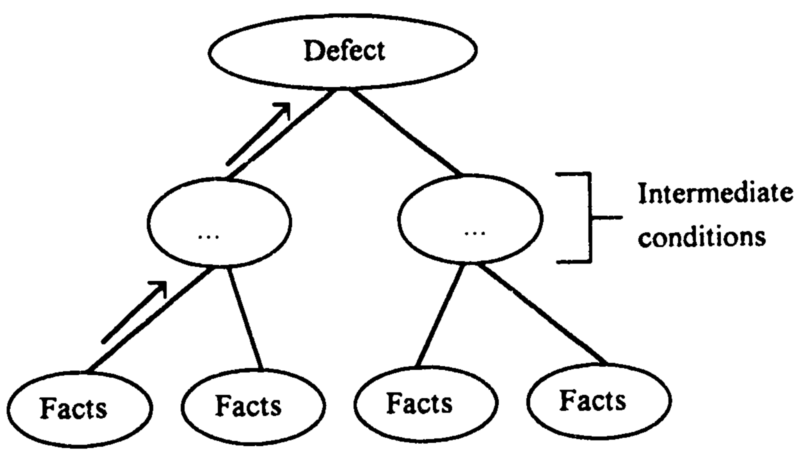

Figure 5.5 Forward chaining reasoning

The opposite is a backward chaining inference engine which receives a statement of conclusion or hypothesis and reasons by checking the facts to verify the truthfulness of the hypothesis. Chapter 6 describes a backward chaining engine in more detail (Alty and Coombs, 1984).

A forward chaining inference engine is more suitable to the task of condition monitoring. This is because the abnormal condition monitoring process is intended to be on-line and therefore receives information from the machine signals only (Milne et al., 1991). Requests for user input are left to the component fault diagnosis section. The VSM™ signals stored in the fact base are the data or facts used by the inference engine which will provide a conclusion as to the state of the machinery. However, the VSM™ signals are limited in number and generally cannot provide an exact conclusion regarding a specific component of the truck. The abnormal condition monitoring only provides a conclusion as to which general area contains the suspected fault, such as the hydraulic or mechanical systems. The component fault diagnosis further narrows the area of failure to a specific component.

5.4 Knowledge base structure for abnormal condition monitoring

Knowledge acquisition is the most important step in building knowledge-based systems. The knowledge collection stage has proven to be the most difficult and time consuming. It requires coordination with the busy schedules of the experts and correlation of their contributions (Nussbaum and Molina, 1992; Hines, 1986). In this project, the knowledge bases used are prototype knowledge bases consisting of a small number of rules to test the functioning of IMSS.

The condition monitoring and fault diagnosis require separate knowledge bases as they are used for separate purposes. It is also useful to have the knowledge bases in smaller units for three reasons. The first reason is that a small knowledge base is easier to modify. The second is that a small knowledge base is more reliable. There is less chance for error or conflict to be entered. Finally, storage and access times are decreased, increasing the reasoning speed.

An examination of the fault tree used for developing the knowledge base for abnormal condition monitoring is as follows (Figure 5.6):

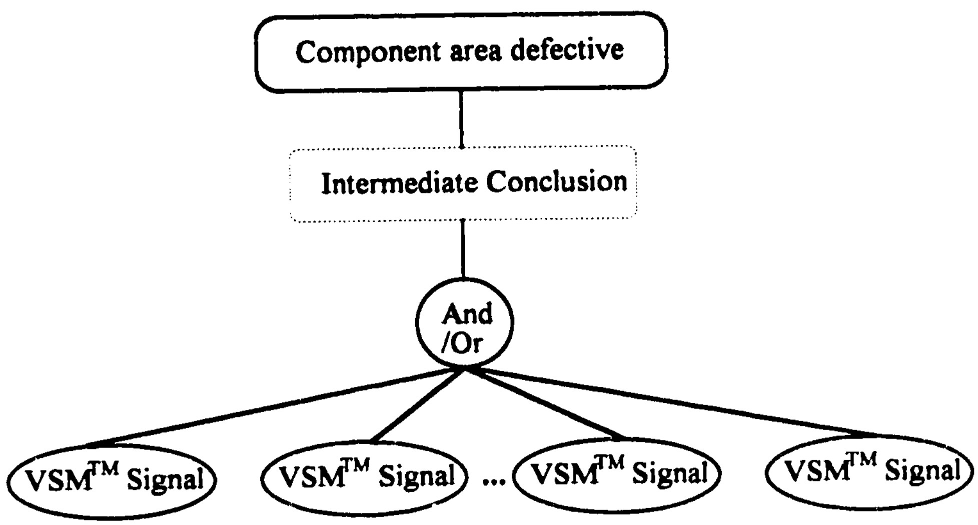

Figure 5.6: Fault tree structure for abnormal condition monitoring

From this diagram, it is seen that the leaf nodes contain the signals from VSM™. The top or root node contains the area of failure that corresponds to a knowledge base developed for the component fault diagnosis module. The intermediate conclusion is included to aid in explaining the reasoning process to the user. This explanation helps the user select a component for analysis in the fault diagnosis area. Table 5.1 contains an example of the rule base for condition monitoring. In the text of the rule base, the VSM™ signals are uppercase, while all other text is in lowercase.

Table 5.1: Condition monitoring knowledge base

```
rule 1: if OIL_PSI < 27
     then mechanical_oil is leaking
rule 2: if mechanical_oil is leaking
     then mechanics are defective
rule 3: if ALTER is FAIL
     then electricity is lo
rule 4: if +24V-MN < 21.5
     then electricity is lo
rule 5: if electricity is lo
     then electronics are defective
rule 6: if ARM1 is OK
     and ARM2 is AL
     then only_one_motor is working_properly
rule 7: if ARM1 is AL
     and ARM2 is OK
     then only_one_motor is working_properly
rule 8: if only_one_motor is working_properly
     then electrical_drive_system is defective
rule 9: if HYDRL is ALM
     then hydraulic_oil is
     insufficient_to_pressure_system
rule 10: if hydraulic_oil is
     insufficient_to_pressure_system
     then steering_hydraulics are defective
rule 11: if STRPRS is LO
     and HYDRL is OK
     then pressure_transmission_system is defective
rule 12: if pressure_transmission_system is defective
     then steering_hydraulics are defective
```

It can be seen that for some rules, such as rules 1 and 2, the use of intermediate conclusions seems excessive. There is only one leaf node associated with one root node. An intermediate conclusion is not necessary for the proper functioning of the inference engine, but is necessary for considerations of information interpretation for the user, helping him understand the reasoning process of the expert.

The signals or facts may be joined in a combination of AND's and OR's, depending on the nature of the knowledge. The IMSS rule structure indicates AND's directly in the rule, as in rules 6, 7, and 11, while indicating OR's by separate rules. Rules 10 and 12 are examples of two separate causes joined by an OR condition to the same conclusion. These separate rules can be joined into one sentence such as "If hydraulic oil is insufficient to pressure system OR pressure transmission system is defective, THEN steering hydraulics are defective."

As stated, the area indicated in the conclusion is the key for indicating the knowledge base for the fault-diagnostic component of IMSS. For example, rule 12 indicates that the fault diagnosis should access the knowledge base for diagnosis of the steering hydraulic system.

Note that the rules have a 3 item sentence structure of name, relation, and value. This standard format, used for representation of facts, as explained in the Data Processing chapter, Section 4.4, is the same format used for representation of the rules. Similar to the compiling of facts, the compiler puts rules into a defined structure and stores them in a binary file with a [***.KB] extension, indicating a knowledge base. The data structure for representation of the knowledge base is shown below:

Table 5.2: KB internal data structure in C

```
union VALUE
{
   char progn[40];
   float fltvb;
};

struct PRIM  /*   premise   */
{
   char if-name [80];
   char if-rela [5];
   int if-type;
   VALUE if-value;
   PRIM *if-nxt;
};
```

Table 5.2: KB internal data structure in C (continued)

```
struct ACTION   /*   action   */
{
   char then-name [80];

   char then-rela [5];
   int then_type;
   VALUE then-value;
};

struct RULE     /*   rule   */
{
   PRIM *ift;
   ACTION *thent;
};
```

The `PRIM` structure contains the "If" conditions in the rule, while the "Then" component is stored in the `ACTION` structure. A combination of "If" and "Then" statements make up a rule, which the compiler stores in a `RULE` structure. The storage of names, relations and values is similar to the storage of facts. The name is stored in `if-name` and `then-name`, relation in `if-rela` and `then-rela`, and the value stored in the appropriate component of the `if-value and then-value` structures. The `*ifnxt` pointer in the `PRIM` structure is used in the creation of a linked list to represent the "And" statements in the rule. An additional component of the structures is the `if-type` and `then-type` integers which indicate the position of the statement in the fault tree. The `then-type` in a root node will have a value of "1" and the `if-type` in a leaf node will have a value of "-1", otherwise, they are assigned a value of "0".

The user employs a text editor to enter the rule base and store it in an ASCII file with the appendage [***.RUL]. Care must be taken when entering rules that the names of the variables and relations exactly match those in the fact base, as the engine operates using textual comparison.

With the standardization of the rule base and condition monitoring in this format, Syncrude may apply IMSS for use with hardware condition monitoring systems other than the VSM™. A new condition monitoring system would require a new rule base and a new data translation scheme, but the basic methodology and programming of IMSS would remain unchanged.

5.5 Implementation of the abnormal condition monitoring inference engine

At present, the abnormal condition monitoring is off-line. A session is initiated by the user from the main menu of IMSS. When the user chooses the "Abnormal" option from the sub menu of "Condition Monitoring" in the "Troubleshooting" menu item, IMSS automatically loads the "Monitor" rule base. Along with this rule base is a fact base of the same name that the "Convert" function created. The forward chaining inference engine reasons through the rule base until it reaches a true condition. The abnormal condition monitoring displays the result on screen and gives the user the option to trace the reasoning process. If "Yes" is selected, the steps in the rule base leading to that conclusion are displayed (Figure 5.7).

The menu of the "Condition Monitoring" window contains two options, "File" and "Link", as seen in the figure. "File" is merely an exit feature, while "Link" can call the component fault diagnosis, described in Chapter 6, for further analysis of the fault. The user may, rather than going to the diagnosis option, review a history of previous cases for that failure area. He or she does so by using the "Incident Report" option of the "Link" menu. The incident reports are a hypermedia feature described in Chapter 7. This linking facility is a part of the methodology of integration for IMSS.

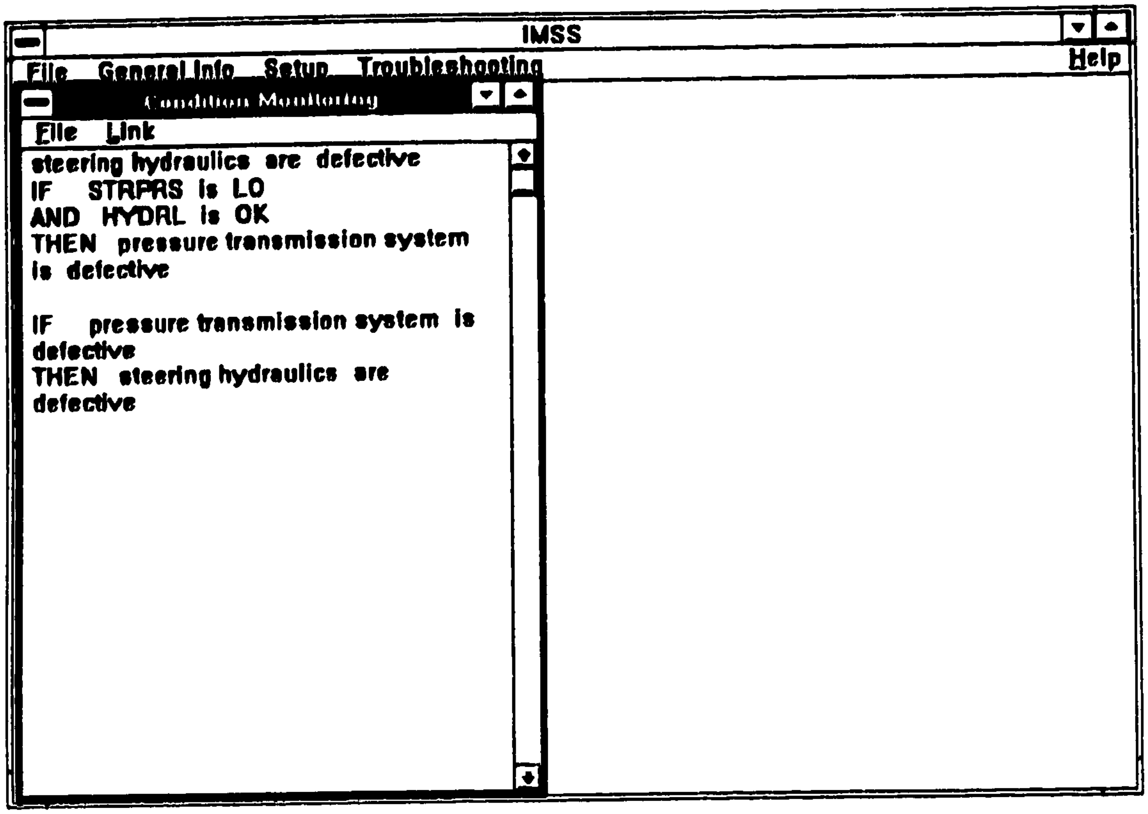

Figure 5.7: Display of abnormal condition monitoring

6 Component Fault Diagnosis

6.1 Purpose and functions

With the complexity of the equipment in consideration, it becomes difficult for any one person to fully understand every component in the system. There may be experts in different areas of the truck structure, but all may not be in the shop on a given day, or they may leave the company and take their knowledge and expertise with them. The component fault diagnosis module has been developed to use the knowledge of the experts in diagnosis via the expert system. It will be a benefit to the experienced employees to have the combined advice of previous experts to assist in diagnosis as a support or second opinion to their decisions. New employees will especially benefit from the intelligent diagnosis aid to make up for their lack of experience. Having this module in place will greatly reduce guess work involved in diagnosis.

The component fault diagnosis module uses an expert system which reasons using VSM™ data and user input, and various hypermedia aids to provide additional information regarding the faults and user tests.

6.2 Component fault diagnosis inference engine based on backward chaining reasoning

The component fault diagnosis inference engine is of the backward chaining type. A backward chaining inference engine is one which receives a hypothesis and verifies the truthfulness of that hypothesis against data or facts. In terms of a fault tree, a root node is the hypothesis. The inference engine traces down the fault tree to verify the conclusion in the root node against the facts in the leaf nodes (Figure 6.1). Forward chaining reasoning was explained in the previous chapter detailing the abnormal condition monitoring.

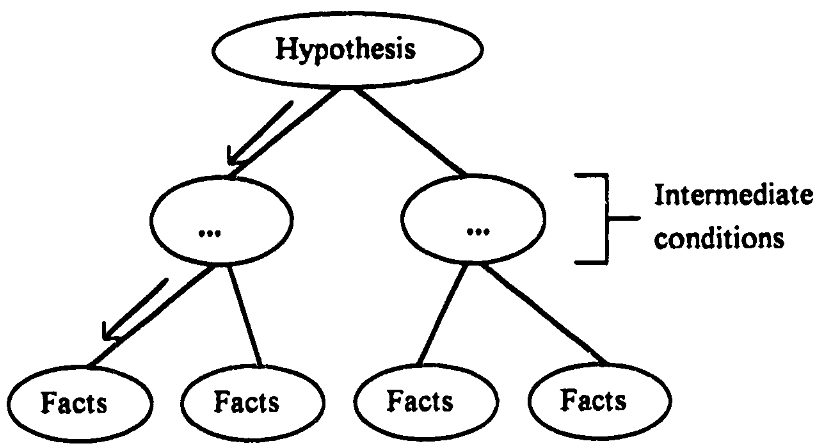

Figure 6.1: Backward chaining reasoning

In the IMSS implementation, the user is given the ability to forth a hypothesis of a faulty component and the reasoning mechanism checks that conclusion against the VSM™ signals and user observations. A backward chaining methodology is suitable for the diagnosis, as there are many possible facts that have to be considered in the diagnostic process, most of which are user observations and tests. If an inference engine were of the forward chaining type, the user would have to fill all the fact nodes to come to a conclusion. This, of course, would lead to useless expense of effort for the worker. It is therefore necessary to reduce the search space of the fault tree by selecting a hypothesis, which would narrow the search to the direct descendants of that node.

6.3 Knowledge base structure for component fault diagnosis

The fault tree for the component fault diagnosis module is similar to that for the abnormal condition monitoring. One of the differences is the root node. Where the condition monitoring gives only the general area of failure, due to the limited specificity of the diagnostic capabilities of the VSM™ signals, the fault diagnosis gives the specific component that has failed. This component corresponds to a component specifically listed and described in the on-line manual. The conclusion of the fault diagnosis is in the format "<component name> is defective." The intermediate node contains the fault that has occurred in that component. The leaf nodes may still contain VSM™ signals, but they primarily contain user tests and observations of the truck components. An outline of the component fault diagnosis fault tree follows:

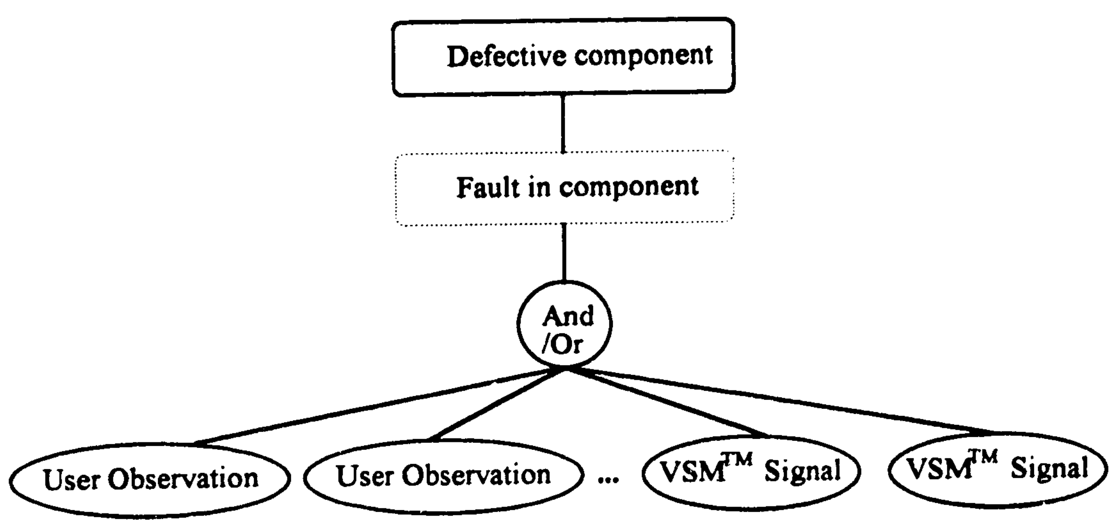

Figure 6.2: Component fault diagnosis fault tree structure

The following table shows an example of the fault diagnostic knowledge base in this fault tree format for the hydraulic area.

Table 6.1: Rule base for component fault diagnosis

```
rule 1: if STRPRS is LO
     and HYDRL is OK
     and steering_pump_hoses are properly_connected
     and relief_valve is properly_set
     and scavenger_blower is operating_correctly
     then steering_pump_compensator is set_improperly

rule 2: if steering_pump_compensator is set_improperly
     then steering_pump is defective

rule 3: if STRPRS is LO
     and steering is slow
     and orbitrol_valve is functioning_properly
     and steering_cylinder is not_damaged_or_plugged
     and priority_spool_bias_spring is undamaged
     and pressure_filter is clean
     then steering_pump is running_slow

rule 4: if steering_pump is running_slow
     then steering_pump is defective

rule 5: if steering_action is not_stopping
     and hoses are properly_connected
     and air_in_steering_lines is bled
     and steering_linkage is not_damaged_or_worn
     then orbitrol_valve is worn_or_stuck

rule 6: if orbitrol_valve is worn_or_stuck
     then orbitrol_valve is defective
```

Table 6.1: Rule base for component fault diagnosis (continued)

```
rule 7: if steering is hard
     and pump is operating_properly
     and relief_valve is not_sticking
     and hydraulic_lines are not_plugged
     and steering_linkage is undamaged
     then orbitrol_valve is
     contaminated_or_improperly_assembled

rule 8: if orbitrol_valve is
     contaminated_or_improperly_assembled
     then orbitrol_valve is defective
```

Rule 1 shows how the component fault diagnosis can derive some of the facts from the VSM signals (STRPRS and HYDRL) while others are from the user observations. Here again, as with the condition monitoring rule base, the AND's are directly part of the rule, and the OR's are indicated by separate rules.

The rule base for the component fault diagnosis is compiled in exactly the same manner as the rule base for condition monitoring, using the "Compile" option from the "Setup" menu.

6.4 Implementation of component fault diagnosis inference engine

The component fault diagnosis module may either be called through the condition monitoring by the "Link" option, or independently from IMSS's main menu through the "Fault Diagnosis" item in the "Troubleshooting" menu. When IMSS opens the fault diagnosis window on the screen, a small dialog box displaying a list of the available knowledge bases appears from which an appropriate knowledge base is selected. The knowledge base selected will correspond to the area of failure indicated by the abnormal condition monitoring. In the fault diagnosis process, the user enters the hypothesis of the failed component into the inference engine. This is accomplished using the list box shown in Figure 6.3.

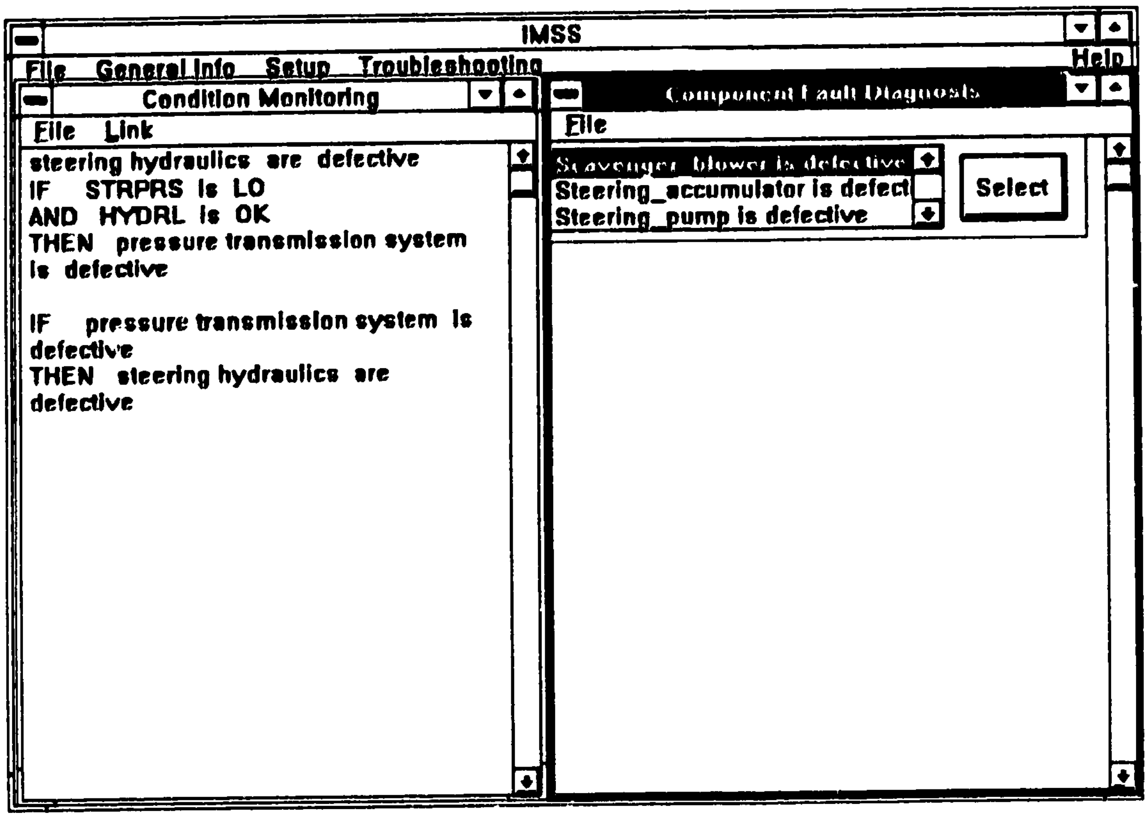

Figure 6.3: List box of available components for analysis

IMSS fills the list box with the result of a scan it performs on the rule base for all the root nodes in the fault tree. A `then-type` value of "-1" in the compiled rule structure identifies the root nodes, as explained in Chapter 5. The user selects the hypothesis using his suspicion of the failure along with the intermediate explanation from the abnormal condition monitoring fault tree. An example of this hypothesis would be, from rule 2, "steering pump is defective." The user may have decided to choose this hypothesis as the intermediate result of the abnormal condition monitoring was "pressure transmission system is defective." As well, he may be aware of some other fact, such as the steering pump not being serviced for several months.

The system needs information from the user when no facts in the fact base correspond to the rule statement. If this is the case, it will bring up a dialog box asking if the statement is true. The user may either respond with "Yes" or "No", or may request "Help" (Figure 6.4).

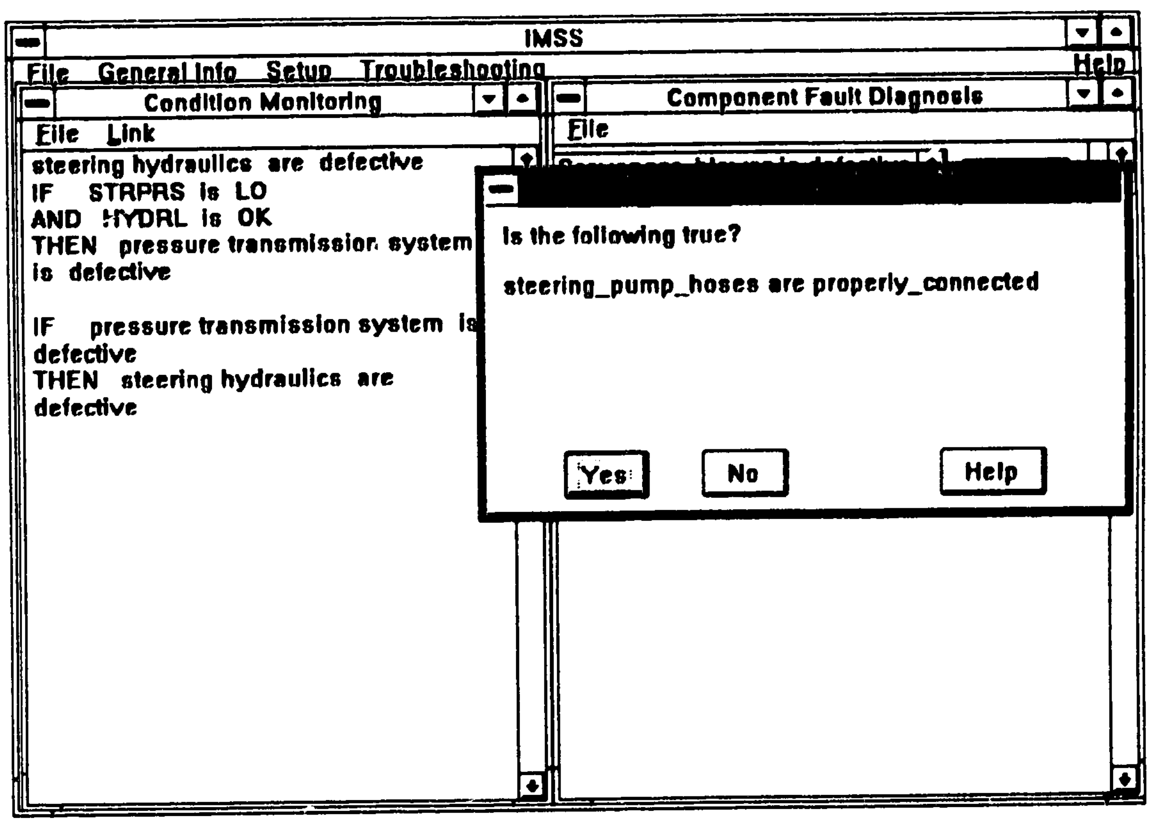

Figure 6.4: Operator input dialog box

If the user presses the button for help, the IMSS transfers to the hypermedia system, which further explains the question (Figure 6.5).

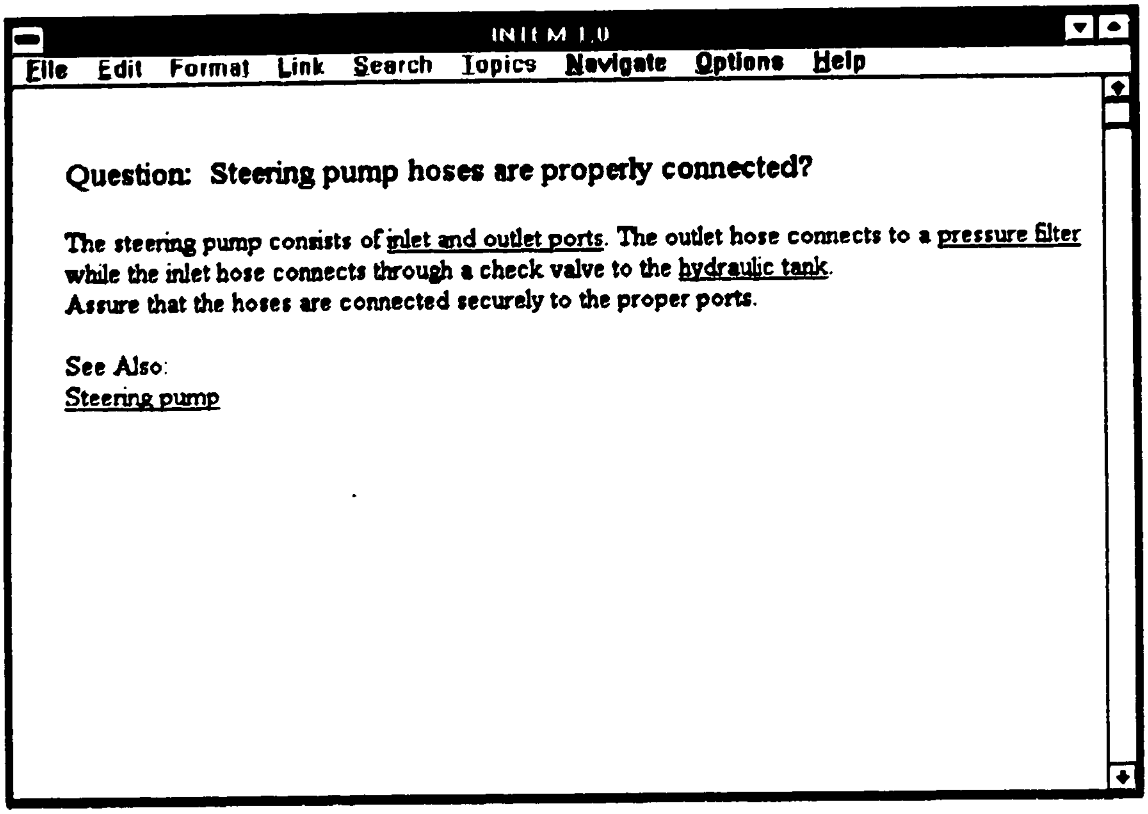

Figure 6.5: Hypermedia help utility for fault diagnosis

More information can be brought to the screen, for example further information about the pressure filter, by clicking the mouse on the underlined words in the text.

The hypermedia document designed for the help function uses the text of each question as a topic name. The topic name is the name of the hypermedia page that is to be displayed. The maximum length of any topic name in the hypermedia system may only be forty characters, while the length of both the question name and the question value may be up to 40 each characters in length. Therefore the question text must first be processed before being sent to the hypermedia system. This is accomplished by parsing the question and taking the first nineteen characters of the question name and joining it with the first nineteen characters of the question value. If the name is shorter than 19 characters, the whole name is used. The same case applies to the question value, if it is shorter than 19 characters. The name and question value are joined with an underscore character. The hypermedia facility and how it is accessed is explained in more detail in Chapter 7.

If the user answers all questions in the affirmative, and the information from the VSM™ corresponds to the conditions in the rule base, IMSS states that the hypothesis is true. It can then display the reasoning path behind the conclusion, which is a restatement of the rules (Figure 6.6).

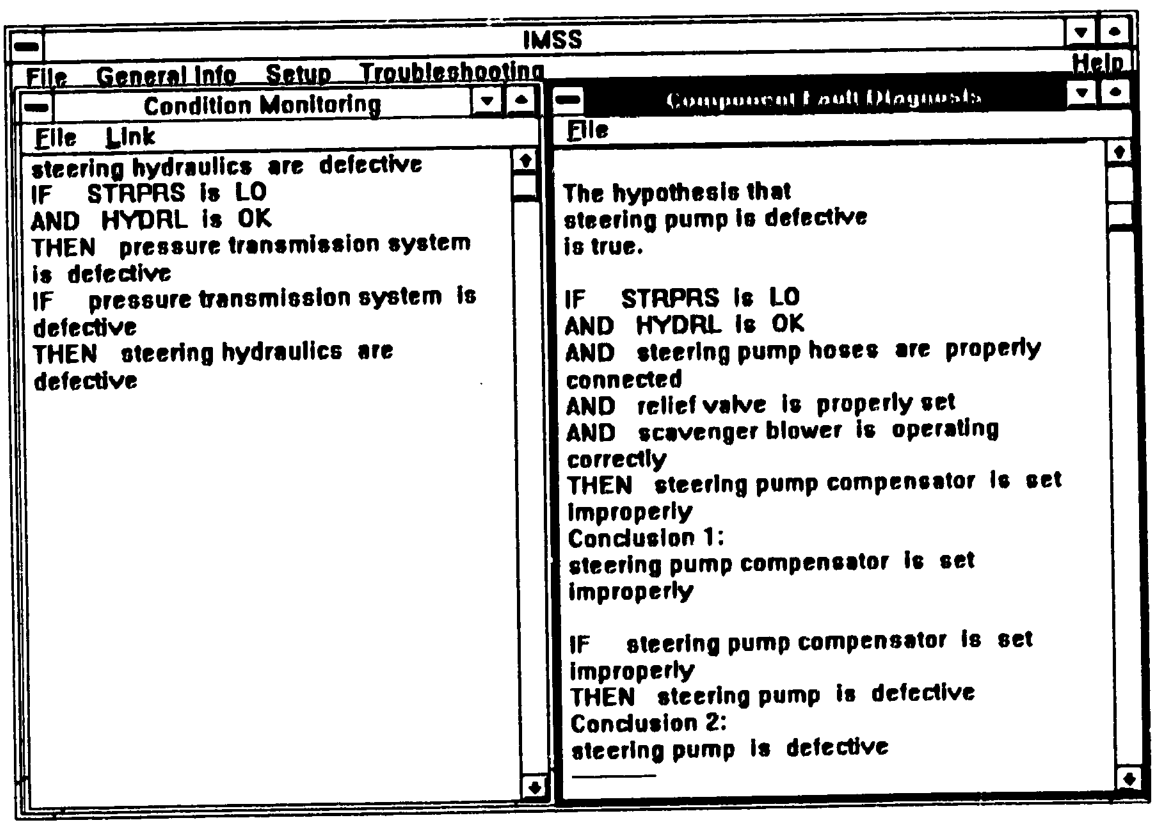

Figure 6.6: Display of reasoning process

From the verification of the failed component by the fault checking process, IMSS has two possible routes for assistance that the user may select from the fault checking menu. They may either go to the fault record on the fault described in the intermediate stage of the fault tree, or they can go to an on-line manual for information regarding that component.

If the option for fault record is chosen, IMSS passes the name and value of the intermediate stage of the fault tree to the hypermedia system which will display a report on that fault (Figure 6.7).

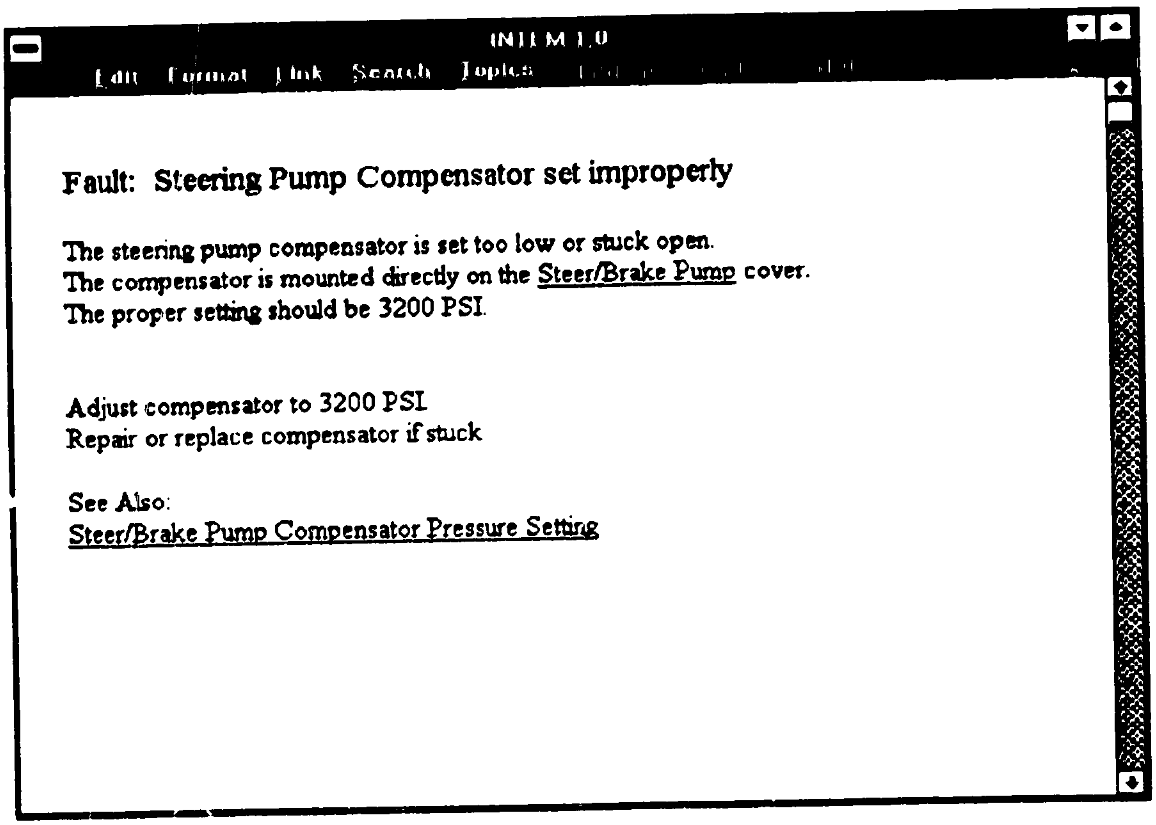

Figure 6.7: Fault record

The report contains a description of the normal state of the component as well as information regarding the remedy for that specific fault. The method of passing the fault to the hypertext system is the same as that used for passing the question to the help system.

Alternatively, the user may call the on-line manual. When calling the on-line manual from this location, the first page shown is the page describing the defective component (Figure 6.8).

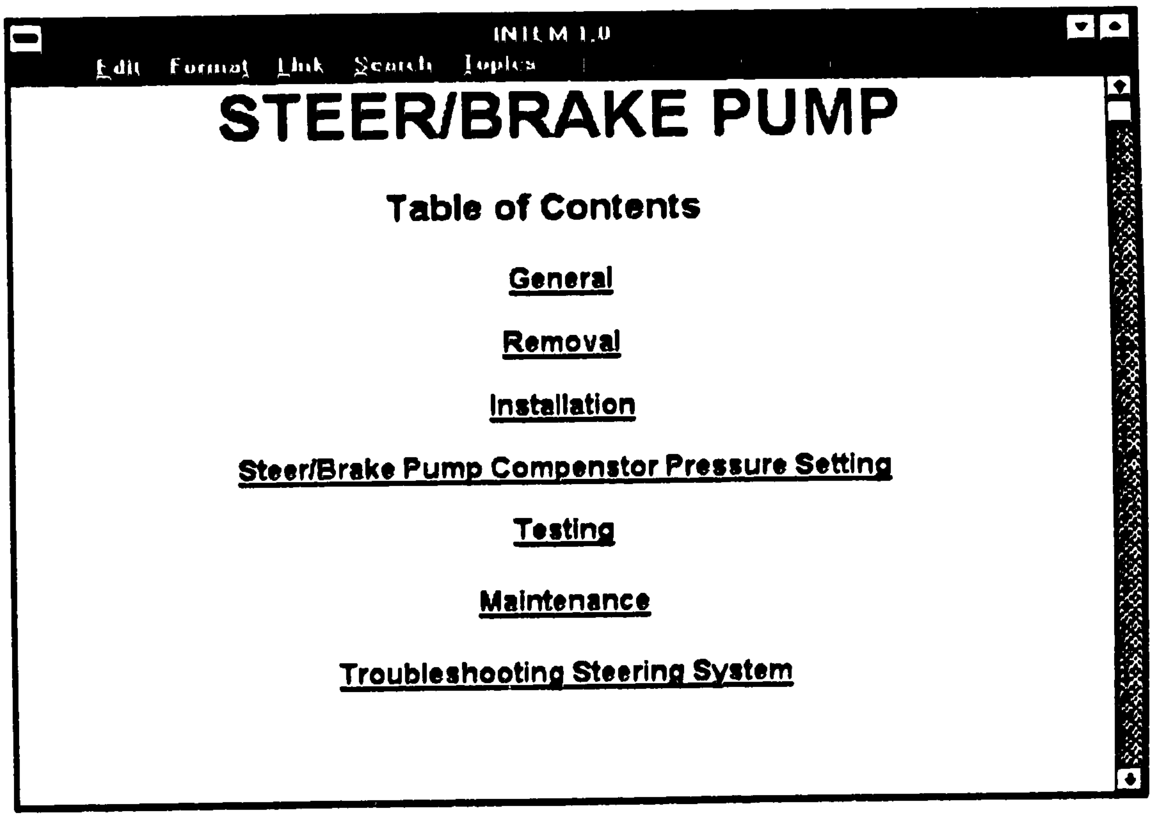

Figure 6.8: Manual page

The topic name for the manual page is the same as the name in the hypothesis, "steering pump," in this case. As the maximum length of the name in the fault checking knowledge base is only 40 characters, the text of the hypothesis requires no additional processing.

There is a significant difference between these two forms of information presentation. The fault record provides information prepared specifically for IMSS and the fault diagnosis. The on-line manual simply gives general information regarding the component as described in the maintenance manuals.

7

Maintenance assistance

7.1 Purpose and functions

In an integrated system, it is useful and important to have on-line advice components to supplement the monitoring and diagnostic components of IMSS. Hypermedia is an excellent tool for this, allowing the user to browse text and graphics in any fashion or having it accessed as necessary by the software. Having the manuals on-line allows the user to have direct and instantaneous access to the necessary page of information. IMSS selects this information intelligently from the results of the reasoning process. There are non-linear links within the hypermedia that allow the user to jump between topics instantaneously backwards and forwards. If the volume of the text is kept reasonable, down to at most one or two pages per topic (Duncan, 1993), the hypermedia tool will be efficient time-wise and a benefit to the user. The hypermedia system also allows for updates, corrections, and user heuristics to be placed in the manual simply by modification of the hypermedia files on disk.

IMSS uses an in-house intelligent hypermedia system for development and implementation of on-line maintenance manuals, incident reporting, outlining of the truck structure, and as a help utility.

7.2 Intelligent hypermedia system

Hypermedia systems allow nonlinear, direct access to vast amounts of information on a variety of subjects as shown in Figure 7.1. Hypermedia is useful in organizing heavily referenced information in a more efficient and effective way.

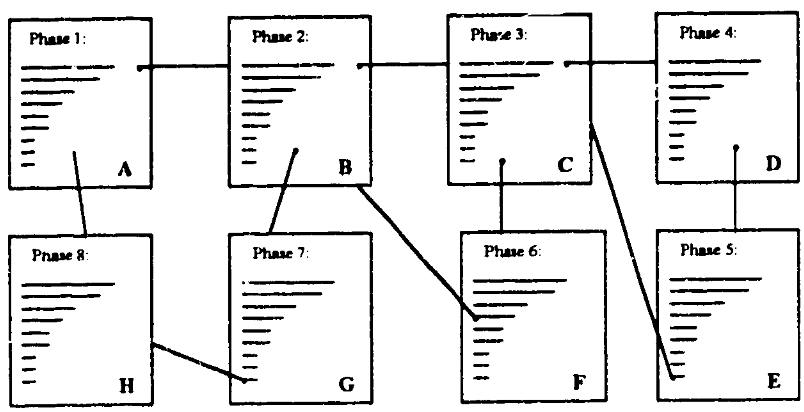

Figure 7.1: Nonlinear retrieval of hypermedia

In a hypermedia system, information is organized into discrete blocks or nodes of information. The blocks of information are then linked in a contextual manner to form the document as a whole and the documents to form an information management system. A link is formed by connecting a key item of information to another related block of information entered in the hypermedia system. Hypermedia technology has been widely applied in engineering design, manufacturing, process control and production management (Abidi and Freeman, 1991; Brauer and Stuchly, 1991).

Most hypermedia systems are useful in developing an information management system. They can provide information and help using various media such as text, graphics, sound, animation, and video. To deal with real-time industrial processes and on-line information management, however, a hypermedia system must have additional capabilities. It must be able to apply formulae to real-time information, and reason from the results using human knowledge and expertise.

To answer to this need, an intelligent hypermedia system (IHS) was developed based on hypermedia, knowledge based systems, and object-oriented programming. This hypermedia system is the one implemented in IMSS. At present, IHS is being used for its information retrieval capabilities, only. IMSS uses separate inference engine modules for the intelligent analysis of the condition monitoring data. The inference engines access the hypermedia system to provide information to the user.

Though IMSS uses none of the intelligent properties of IHS, it was decided to use this hypermedia system in the expectation of future utilities. One objective of IMSS was to provide an integrated system. An intelligent hypermedia system would provide the opportunity for interactive form generation, procedural checklists, and intelligent updating of manuals and records, and allow flexibility for other needs that may arise in the future.

7.3 Communication of IMSS with IHS

The hypermedia system is a separate program which IMSS system accesses. Before calling the hypermedia system, IMSS writes a string to a file named "CMDLINE.@@@." This file contains a single string with three words separated by spaces. The first word is the name of the hypermedia program, "HYPER1.EXE" so the hypermedia system knows that the information in the file is intended for its use. Next in the string is the name of the hypermedia file, such as "TITAN", the name of the on-line manual file. Following that is the page or topic name of the hypermedia file that the IMSS requires. If the first page of the hypermedia file is being accessed, the topic name is typically "ROOT". Therefore, to open the on-line manual to the first topic, which in this case is the title page, would require the string "HYPER1.EXE TITAN ROOT" to be stored in the "CMDLINE.@@@" file. To access a specific topic or page within that file, the word "ROOT" would be substituted by the topic name for that page.

When IMSS needs to open IHS, it first checks to see if it is already running by using a call to the FindWindow function in the Microsoft Windows™ API (Anon., 1991). At present, the FindWindow function just looks for the text in the title bar of the IHS window, as the hypermedia system does not yet have a class name. If IMSS locates it, it brings the hypermedia window to the front using the SetFocus function. If the window is not found to be already running, the hypermedia system is called by the WinExec function. To return control to the IMSS system, the user may either minimize the IHS window or switch the focus by clicking on the IMSS window, as they would to escape any Microsoft Windows™ help function.

An example of the code used to communicate with the IHS as described follows in Table 7.1.

Table 7.1: IMSS communication code with the IMSS

```
void TImssWindow::CMOpenManual(RTMessage)
{
     //  Load the hourglass wait cursor
     HCURSOR HCursor;

     HCursor=LoadCursor(NULL,IDC_WAIT);
     SetCursor(HCursor);

     //  Write the command string to the file cmdline.@@@
     FILE *fp=fopen("cmdline.@@@","w");
     fprintf(fp,"hyper1.exe titan root");
     fclose(fp);

     //  Look if the window is already running
     wdw=FindWindow(NULL,"INTEM 1.0 ");

     //  If not, run the hypermedia system
     if (wdw == NULL)
          WinExec("hyper.exe", SW_SHOWNORMAL);

     //  If it is, bring it to the front
     else
     {
          ShowWindow( wdw , SW_SHOWNORMAL);
          SetFocus( wdw );
     };
}
```

7.4 Hypermedia tools in the maintenance assistance

As stated, the maintenance assistance module provides such hypermedia tools as on-line maintenance manuals, incident reporting, truck structure detailing, and help utilities. These functions are described in detail, here, with associated figures.

7.4.1 On-line manual

The manuals for the truck are being put on line. It has already been seen that the component fault diagnosis module can access the manuals, calling a specific topic from the manual related to the defective component. The user may also directly access the manual using the main menu of IMSS through the "Open Manual" item in the "General Info" menu. The manual appears in the hypermedia system open at the first page (Figure 7.2). From there, the user may browse through all available topics in the manual.

Figure 7.2: Maintenance manual title page

7.4.2 Incident reports

When a human comes across an error in a system, often the first thing he will do is reflect if this situation has occurred before and how it was remedied. The purpose of the incident report is to record occurrences of a fault to make it available for future review. If the incident reports are stored using the hypermedia system, they can be accessed intelligently by IMSS. According to the architecture of IMSS, when the abnormal condition monitoring determines a fault area, the "Link" menu gives the option to view the incident reports. If this option is chosen, IMSS goes to the incident reports for that fault area. Just as there is a fault diagnostic knowledge base for the fault area, there is a group of incident reports as well.

Similar to the other calls to the hypermedia, IMSS passes the name of the fault area to the hypermedia system. The hypermedia displays the topic with the name corresponding exactly to the name of the fault area for the user to browse. An example for the steering system is shown in Figure 7.3. As well, the user may view the whole incident report file from the first page by selecting the "Incident Report" option from the "Troubleshooting" item in the main menu.

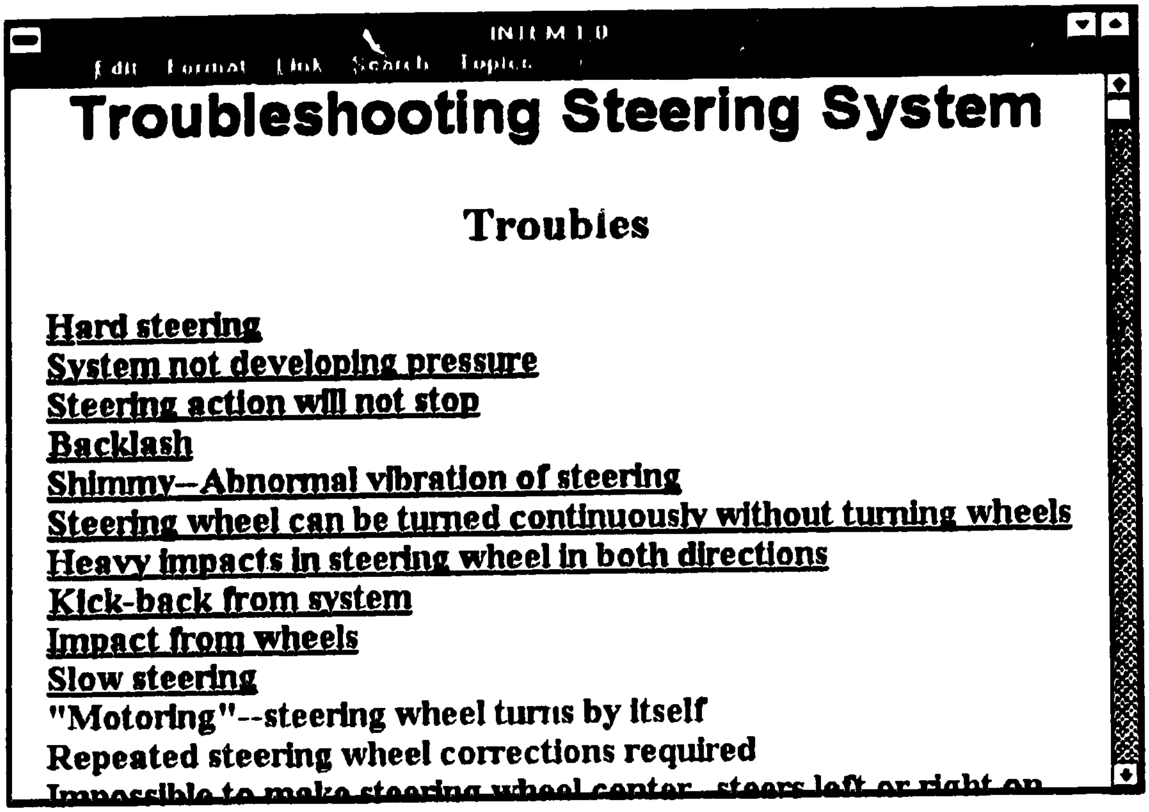

Figure 7.3: Incident reports of steering system

7.4.3 Truck structure

The user may graphically browse the structure of the truck to gain an understanding of how the various components in the system interconnect. The purpose of this function is mainly for information and reference. Figure 7.4 shows the overall layout of the fluid power structure.

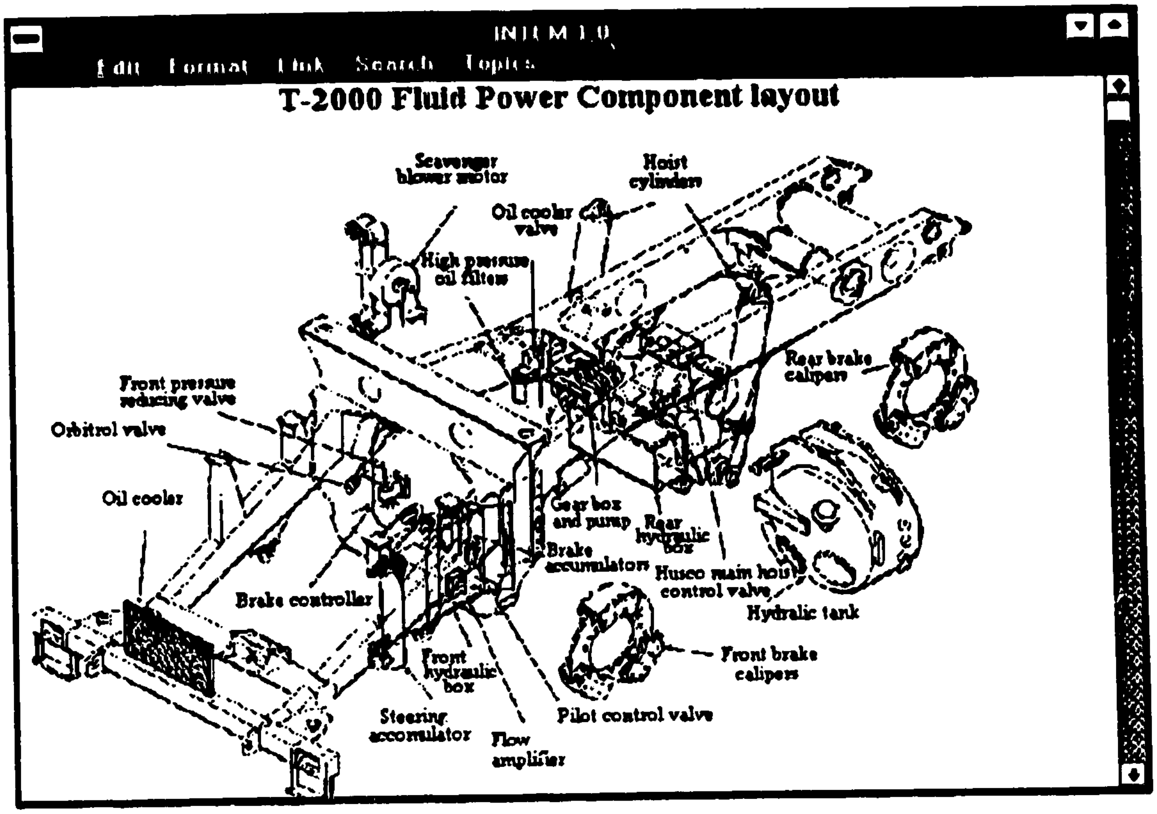

Figure 7.4: Overall layout of fluid power structure

Moving the mouse cursor over any of the named components and clicking on the name takes the user to a textual description of the component along with expanded graphical views (Figures 7.5 and 7.6).

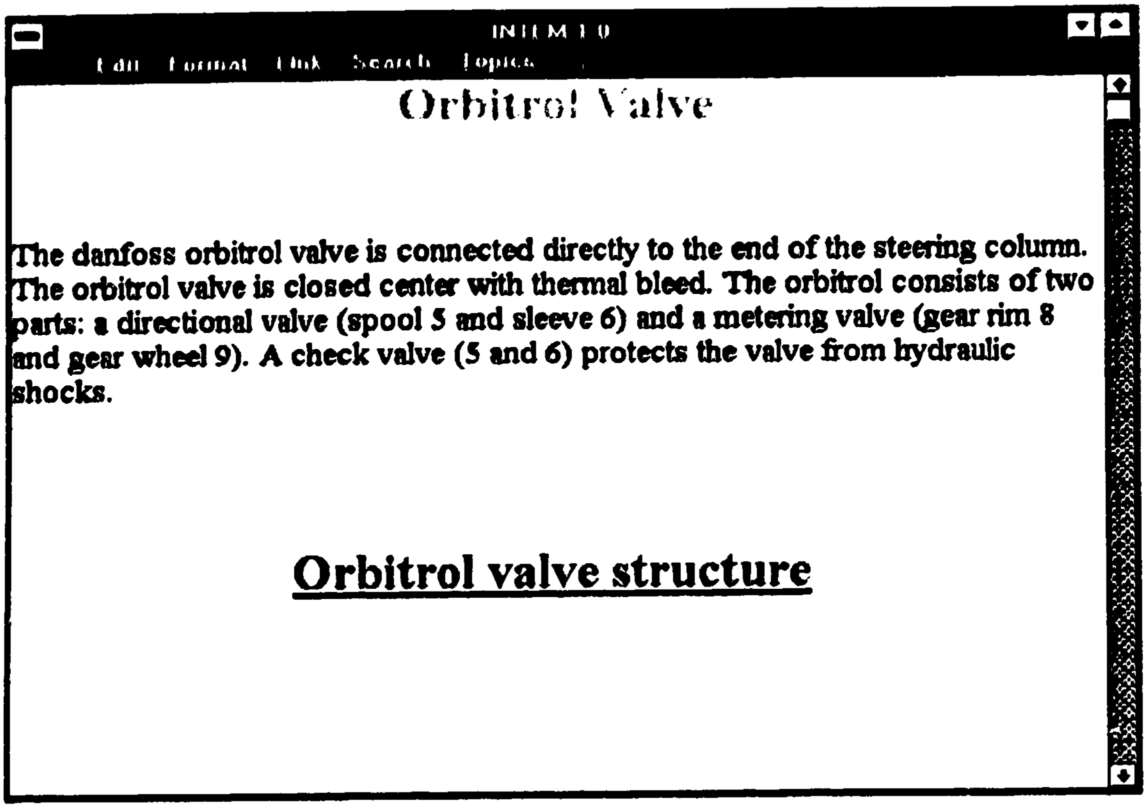

Figure 7.5: Obitrol valve description

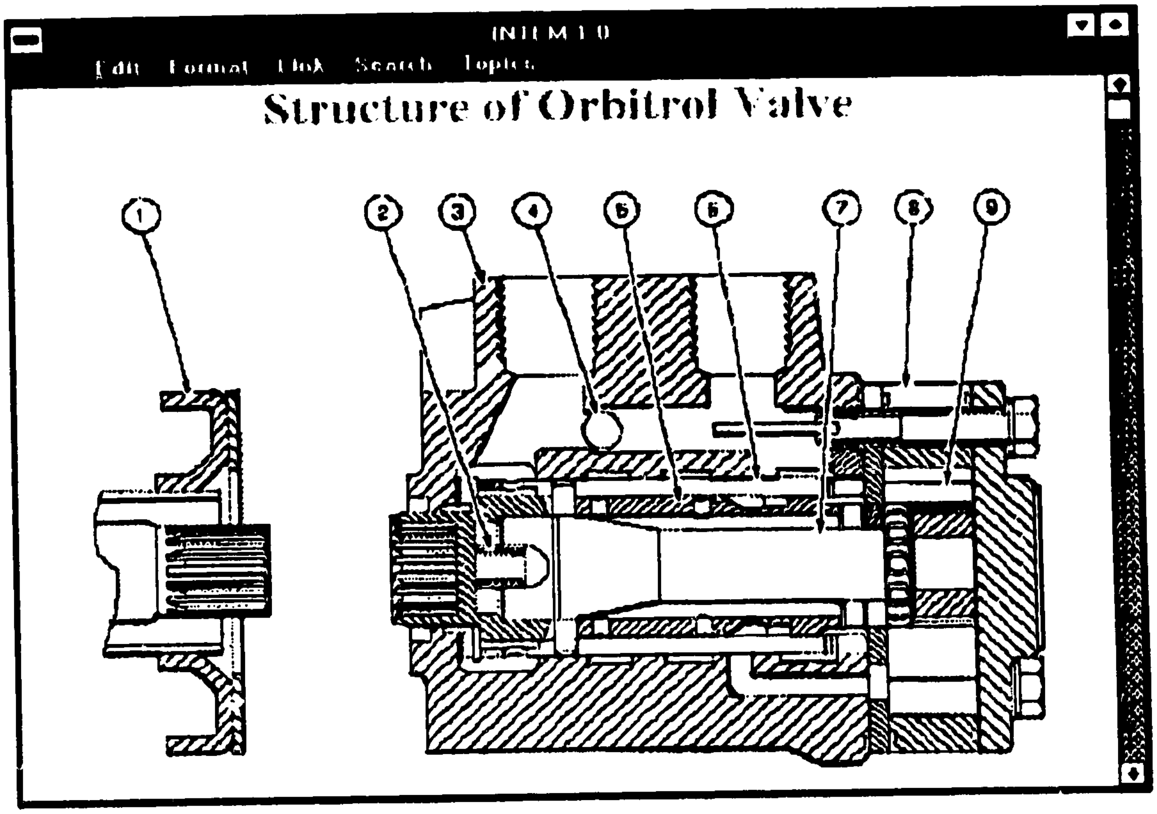

Figure 7.6: Orbitrol valve picture

Using this method the user will be able to navigate the various structures such as mechanical, electrical, hydraulic structures, of the mining truck. At present only a portion of the hydraulic system has been completed. This structure browse option is called through the "Structure" item in the "General Info" menu.

7.4.4 Help utilities

Under the "Help" menu there are two options namely "Configuration" and "Commands". Both of these make use of the hypermedia tool. The "Configuration" option will provide information on the functional process of IMSS, but is yet undeveloped.

The "Commands" option gives a description of the various functions available through the menus of IMSS. A diagram (Figure 7.7) shows the outline of the IMSS menu. The user may move the mouse cursor over any of these blocks in the diagram and click on the box to obtain further information regarding that menu item.

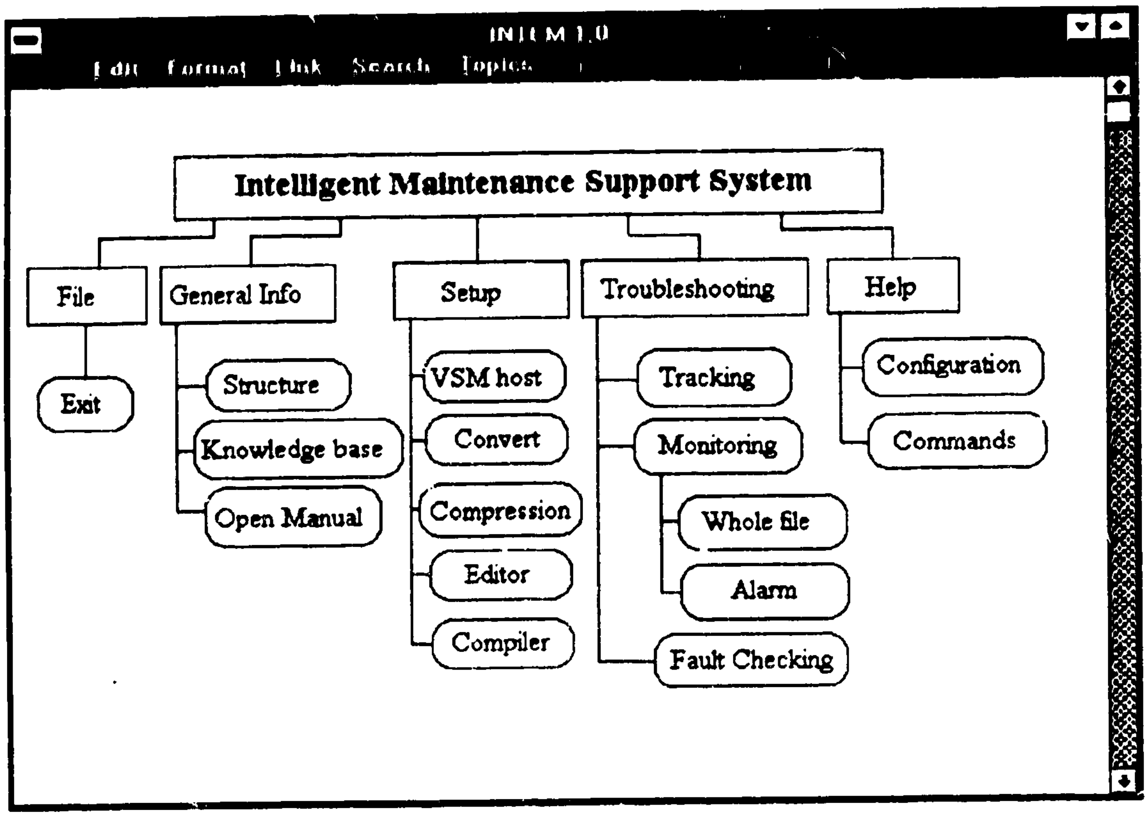

Figure 7.7: Help menu structure

8

Conclusions

8.1 Summary

An Intelligent Maintenance Support System (IMSS) for truck condition monitoring has been successfully developed. It has not been commercially implemented yet but several test runs have been done and it has performed effectively. The important conclusions are:

1) A practical methodology for truck condition monitoring, fault diagnosis, and maintenance assistance has been developed. This involves monitoring of condition data by a software condition monitoring system. When a fault area is determined, control is passed to a diagnostic component that either works from historical cases or interactively with the user from a diagnostic rule base. When the defective component is determined, the system provides assistance using on-line manuals and fault remedy procedures.

2) In IMSS, both forward and backward chaining inference engines have been implemented. The forward chaining inference engine is data driven, and therefore is suitable for use in interpretation of hardware monitoring signals. The backward chaining

methodology works by verifying a hypothesis against given facts, and is therefore used in the fault diagnostic process, by verifying faults against the hardware signals and user tests and information. As well, the forward chaining monitoring acts to reduce the search space of the backward chaining diagnosis.

3) The IMSS software, using the above methodology and inference engine structure, was developed for use with the Titan T-2000™ series trucks at the Syncrude oil sands mine. It is still in the prototype stage with simple knowledge bases developed for condition monitoring and diagnosis of the hydraulic and hoist systems. The software to display the trend of the monitored variables was also developed. Some hypermedia files were developed for maintenance assistance specific to the conclusions available through the current knowledge bases. Although this system was developed for use with the Vital Signs Monitor™ hardware condition monitoring system, it should be able to work with other systems with a modified data conversion interface.

4) An intelligent hypermedia system is used for maintenance assistance purposes. The overall referencing process will be speeded up with the indexing capabilities of the hypermedia system. There are other hypermedia aids in the maintenance assistance functions to assist both the new and the experienced employees.

5) Data handling functions have been developed to interface between the hardware condition monitoring data and IMSS. The data from the VSM™ has to be converted to the three part sentence fact base format for use by the inference engines. As well, the variable data is stored in the binary files for use by the normal condition monitoring unit. It is preferable to have this data in established formats for use by the IMSS inference

engines and other modules rather than having them read directly from the VSM™ files for the following two reasons: first, the data reading and parsing are lengthy and time consuming processes, and second, the system can be adapted to other hardware condition monitoring systems.

6) The user interface was developed using Microsoft Windows™. Microsoft Windows™ is an easily understood system and is a standard in the PC industry. It is presently the preferred platform for software at Syncrude. It provides the ability for the user to view the entire monitoring and diagnostic process in one screen, enabling the user to get a feel for the overall process of IMSS. In terms of developing an integrated system, Microsoft Windows™ allows for a number of programs to be open at the same time, as is the case with IMSS. The normal condition monitoring and the hypermedia application are both executable programs separate from IMSS. The focus may be switched between them as necessary during the functioning of the IMSS.

8.2 Areas of improvement

Through research involvement with this project, it has become apparent that there are some areas in which this project may be further improved.

8.2.1 Frame-based structure

The development of rules using a 3 part sentence structure is not robust enough to accommodate all the indexing features. At present, the hypermedia topic is selected using the first 19 letters of the name and relation. It is possible, as the name and relation may each be up to 40 characters long, that the topic name parsed from this sentence is not unique. In another case it may be necessary to choose a hypermedia topic that does not even correspond to the name-relation string. It would be preferable to have the topic name as a specific slot in the frame containing the rule.

The same problem is found in the automation of knowledge base selection by the condition monitoring. Presently, the user must choose the knowledge base corresponding to the conclusion of the condition monitoring rule. The knowledge base may correspond to the first eight characters of the name of the conclusion of the condition monitoring. This is because MS-DOS™* file names may be eight characters long. Here there is even less chance of uniqueness. Again, the use of a frame structure would resolve this problem.

Sometimes the knowledge base developer is a domain expert and not knowledgeable in the structure of frames or knowledge based systems. In this case, it is preferable to have the knowledge base set up as a simple text file with a series of rules written in plain English. In this application, however, it is likely that the person entering the data will be a computer expert who is entering the knowledge gained from domain experts. He will be familiar with, or be able to learn easily, the frame structure.

---

* MS-DOS is a trademark of Microsoft Corporation

8.2.2 Interface with commercial spreadsheet

At present, the parsed data from the compression process is stored in a custom designed binary file for use by the normal condition monitoring. In the consideration of future flexibility, it would be preferable to have this data in a format readily usable by commercial spreadsheet programs. Microsoft Excel™* is the spreadsheet used most commonly by the Syncrude employees, and is suitable for interface with Microsoft Windows™ applications.

* Microsoft Excel is a trademark of Microsoft Corporation

References

Abidi, A. A., and Freeman, E. M., (1991). "A hypertext system for designing squirrel-cage induction motors", Computer-Aided Eng. 8(5), pp.212-16.

Alty, J. L., Coombs, M. J., (1984). "Expert systems - concepts and examples", NCC Publications, Manchester, England.

Anon., (1991). "Windows API volume 1", Borland International.

Anon., (1992a). "Titan T-2240 truck - maintenance manuals", Marathon LeTourneau Company.

Anon., (1992b). "Boland C++ version 3.1 - library reference", Borland International.

Anvar, A. M., Kuhnell, B. T., (1991). "Expert system for automobile engine diagnosis", International Mechanical Engineering Congress, Sydney, 8-12 July, pp. 58-63.

Bandopadhyay, S., Venkatasubramanian, P., (1990). "A fault-diagnostic expert system for walking draglines", Mining Science and Technology, 11, pp. 71-83.

Billington, R., (1990). "The application of expert systems for health monitoring of machines", Mining Technology, May, pp 143-146.

Brauer, J., and Stuchly, J., (1991). "An information management system for engineering design based on hypertext", Computer-Aided Eng. 8(5), pp.189-93.

Deibort, H. J., Cobb, J. W., Denton, R. W., (1991). "Nethelp user friendly interface", Conference Proceedings - IEEE SOUTHEASTCON v 2, Williamsburg, VA, USA, pp. 746-750.

Donald, D. A., (1989). "Long term mine planning at Syncrude", International Journal of Surface Mining, 3, pp. 27-34.

Duncan, R., (1993). "Building the basic sturcture of a Windows Help file", PC Magazine, May, 11, pp. 349-354.

Granholm, S., Kumar, U., (1990). "Some recent thinking in field of equipment maintenance in mines", Mine Planning and Equipment Selection, pp. 73-78.

Halasz, M., Davidson, P., Abu-Hakima, S., and Phan, S., (1992). "JETA: A Knowledge-Based Approach to Aircraft Gas Turbine Engine Maintencance", Journal of Applied Intelligence, 2, pp. 25-46.

Hines, J. G., (1986). "Collecting information and representing knowledge", IEE Colloquium (Digest), n 1986/59, pp. 2.1-2.2.

House, J. H., (1990). "Development of Titan T-2240 rear dump truck", Mine Planning and Equipment Selection, pp. 193-208.

Hu, David, (1989). "C/C++ for Expert Systems", MIS Press.

Jiao, J., Cha, J., Rao, M., (1993). "IIFDS: An integrated intelligent fault diagnosis system for process industries", Conference on Computer Integrated Manufacturing in the Process Industries, East Brunswick, NJ, USA, pp. 302-316.

Johnson, J. W., (1989). "Application of drive systems for mine haulage vehicles", International Journal of Surface Mining and Reclamation, 3, pp. 155-161.

Jonah, K., (1988). "Syncrude's mine production planning", Mine Planning and Equipment Selection, pp. 443-455.

Kar, T. R., Topuz, E., (1989). "A fault diagnostic method for equipment failures", International Journal of Surface Mining and Reclamation, 3, pp. 195-199.

Karwatzki, J. M., (1987). "Expert systems in engineering", Computer-Aided Engineering Journal, pp. 99-102.

Kumar, U., Rutquist, J., Jadmalm, P., (1988). "The changes in the concepts of maintenance for a large mechanized surface mine and its cost effectiveness: A case study", Mine Planning and Equipment Selection, pp. 237-244.

Landauer, C., (1992). "Integrated software environments", IEEE 8th Conference on Artificial Intelligence for Applications, Monterey, CA, USA, pp. 164-170.

Malhotra, Y. P., List, S. J., (1989). "Short term mine planning at Syncrude", International Journal of Surface Mining and Reclamation, 3, pp. 11-20.

Marciano, J., Pratt, J., Carlton, L., Carlson, R., (1990). "Mechanical system condition monitoring (MSCM) system design", IEEE Autotestcon 90, San Antonio, TX, USA, pp. 543-548.

McKee, W. N., (1989). "Overview of the Syncrude mining operation", International Journal of Surface Mining and Reclamation, 3, pp. 1-9.

McKee, W. N., (1990). "Technical support to the Syncrude oilsands mine", International Journal of Surface Mining and Reclamation, 4, pp. 83-7.

Milne, R., (1986). "Artificial intelligence applied to condition health monitoring", IEE Colloqium (Digest), n 1986/87, pp. 5.1 - 5.4.

Milne, R., (1991). "Predicting faults with real-time diagnosis", Proceedings of the IEEE 30th Conference on Decision and Control, Brighton, England, pp. 2598-2603.

Mitchell, J., (1991). "Diagnostic expert system techniques for improving hydraulic maintenance of a continuous mining machine", Mining Engineering, April, pp. 419-423.

Narazaki, H., Iwatani, T., Omura, K., Otsuka, Y., and Konishi, M., (1990). "An AI Tool and Its Applications to Diagnosis Problems", ISIJ International, Vol 30, pp. 98-104.

Nussbaum, M., Molina, O., (1992). "Intelligent manual: an aid for process engineering", Engineering Applications of Artificial Intelligence, 5(1), pp. 43-49.

Oommen, M., Santa Cruz, C., Kohler, J. L., (1987). "Maintainability improvement of mining machines using knowledge-based expert systems", IEEE Mining Industry Technical Conference, Pittsburgh, PA, USA, pp. 103-118.

Rao, M., Wang, Q., and Cha, J., (1993). "Integrated Distributed Intelligent Systems in Manufacturing", Chapman & Hall, London.

Rawicz, A., Jiang, H., (1992). "Diagnostic Expert-System for Mechanical Reliability in Heavy Trucks", Proceedings of the IEEE Anuual Reliability and Maintainablility Symposium, Las Vegas, NV, USA, pp. 426-431.

Rawicz, A., Xie, Z. W., Rowley, A., (1990). "Multi-probe vibration analysis for reliability", Proceedings of the IEEE Annual Reliability and Maintainability Symposium, Los Angeles, CA, USA, pp. 331-335.

Schildt, H., (1987). "Artificial intelligence using C", Osborne McGraw-Hill, Berkeley, CA, USA.

Scoble, M. J., Peck, J., and Hendricks, C. (1991). "A study of surface mine equipment monitoring", International Journal of Surface Mining and Reclamation, 5, pp. 111-116.

Senjen, R., de Beler, M., Leckie, C., Rowles, C., (1993). "Hybrid Expert Systems for Monitoring and Fault Diagnosis", Proceedings of the IEEE 9th Conference on Artificial Intelligence for Applications, Orlando, FL, USA, pp. 235-241.

Shortliffe, E. H., (1976). "Computer-based medical consultations: MYCIN", American Elsevier Publishing Co. Inc., New York, NY, USA.

Sullivan, T. W., (1990). "New technology and economies of scale in shovel-truck sizing", Mine Planning and Equipment Selection, pp. 3-6.

Ursenbach, A., Wang, Q., Rao, M., Coward, J., Lamb, D. K., (1994a). "Intelligent Maintenance Support System for Mining Truck Condition Monitoring and Troubleshooting", Accepted, International Journal of Surface Mining and Reclamation.

Ursenbach, A., Wang, Q., Rao, M., Coward, J., Lamb, D. K., (1994b). "Intelligent Maintenance Support System for Syncrude Mining Truck Condition Monitoring and Troubleshooting", Submitted, CIM Journal.

Ursenbach, A., Wang, Q., Rao, M., Coward, J., Lamb, D. K., (1994c). "Data Handling for Information from Vital Signs Monitor on Syncrude Canada's Mining Trucks", Accepted, 44th Canadian Chemical Engineering Conference, Calgary, Canada.

Walter, M., (1990). "Mechanical diagnostics - past, present and future", IEEE Autotestcon 90, San Antonio, TX, USA, pp. 517-521.

Waterbury, R. C., (1988). "Expert systems extend expertise", InTech, 35(12), pp. 22-27.

Wing, G., Uttamsingh, T., (1990). "AI Based Diagnostic and Decision Support Systems for the Trucking Industry", SAE Transactions, 99(2), pp. 744-752.

Zhu, X., Wang, Q, Ursenbach, A., Rao, M., Zuo, M., (1993). "Intelligent Maintenance Support System for Syncrude Mining Trucks", Proceedings 1993 Canadian Conference on Electrical and Computer Engineering, Vancouver, Canada, pp. 1217-1220.