Model Based Diagnosis for Network Communication Faults

Leliane Nunes de Barros
leliane@ime.usp.br
IME - BCC
University of São Paulo
Rua do Matão, 1010 São Paulo
SP Brazil - 04717 - 020

Marilza Lemos
mlemos@lsi.usp.br
LSI - EPUSP
University of São Paulo
Av. Prof. Luciano Gualberto,
158
São Paulo - Brazil 05508 - 900

Volnys Bernal
volnys@lsi.usp.br
LSI-EPUSP
University of São Paulo
Av. Prof. Luciano Gualberto, 158
São Paulo - Brazil 05508 - 900

Jacques Wainer
wainer@dcc.unicamp.br
Instituto de Computação
University of Campinas
Campinas -SP - Brazil

Abstract

The lack of specialized professionals in network management and the growing complexity of this task has been aiming the need for developing tools to give support to the network administrator task. The construction of such tools requires an intense process of knowledge acquisition from experts in the area as well as the use of Artificial Intelligence (AI) techniques. A number of different approaches have been proposed, evolving from rule-based systems through case-based systems, to more recent model-based systems [6] [7] [8] [9] [11]. A special attention has been given to propose systems to solve two main network management tasks: the fault diagnosis and performance management. The aim of this paper is to specify a Communication Fault Diagnostic System applying the AI Model Based approach.. We claim that this approach provides a foundation for exchanging behavioral, structural and control information between the sub-tasks of such complex systems. We also show what are the main aspects to be considered when constructing such systems: the construction of an automatic network discovery system and a configuration diagnosis system, both to support the construction of the network configuration model, and a network status gathering system to allow the diagnosis system to observe the network.

1. Introduction

The complexity on building a diagnostic system for Network Management resides on: a regular network can have a variety of types of hardware components and a large number of them; there are different types of software components (protocols, operating systems, services, applications); the equipments and connections may be changed, and yet some network protocols are based on dynamic configuration.

The construction of network models to build management tools involves the identification of all necessary knowledge and its organization in such way that the management task can be automatically performed as an activity of exchanging behavioral, structural and control information.

The Model Based Diagnostic approach (MBD) proposes the construction of Knowledge Based Systems (KBS) through the specification of two basic structures: Domain Models and Problem Solving Methods (PSM) [1] [3] [4] [10] [12] [13] [14]. An uniform representation for network models allows matching of components and variables among different views of the network domain. We claim that Model Based Diagnostic provides a foundation for exchanging information between the sub-tasks of a complex system.

In this work we apply the Model Based approach to perform the communication fault diagnosis in the network domain. Therefore the proposed method consults models that represent the network in its multiple aspects, such as: configuration model, performance model, fault-states causal model, equipment models, and others.

In particular, to construct the Configuration Model we have developed a system for network discovery and construction of the network configuration *(Network Discovery System)*. Since the network configuration is the result of a human activity, errors can be embedded in the discovered configuration model. For that reason we have also developed a *Configuration Diagnosis System* which can detects a set of configuration errors during the acquisition and construction of the network models.

In the next sections we will present different aspects involving in the development of the Model Based Diagnostic System.

2. Identifying and building network models

To construct domain models one should identify a set of terms which can be used to describe knowledge about the domain. A domain ontology corresponds to an organized set of domain generic terms that can be used to describe a particular domain, in this case, the communication network domain. However in order to perform specific tasks in the domain, extra terms should be considered. For instance, the diagnostic task involves terms such as: hypotheses, observations, symptoms, fault, and so on.

This material is based on work supported by the ABC Bull SA Telematic..

The ontology terms can be combined to generate more complex structures involving hierarchies and dependencies such as the Configuration Model described in the next Section. The importance of building a domain ontology for the Management Network task resides on the fact they are skeletons (i) to be used as data structures to model and represent knowledge; (ii) to be instanciated by the specific domain elements during problem solving reasoning and (iii) to be used as *knowledge roles*, e.g., the input/output of subtasks of a method (roles that domain terms can play during problem solving, i.e., a Hub can play the role of hypotheses and later become the diagnostic).

Table 1 shows a part of an ontology for the Network Management task organized in a taxonomy of knowledge roles, focusing on the diagnostic task. We have identified two basic types of knowledge used by the human manager while performing the diagnostic task: the *Network Knowledge* and *Network Management Knowledge*. The *Network Knowledge* contains all the information about the network environment such as the Configuration Model which is composed by the configuration levels described bellow (Section 2.1). The *Network Management Knowledge* is composed by the terms essentially used to describe network management knowledge including diagnostic knowledge, such as Network Performance Model, Network Fault States Model or yet the Network Path Model that will described in sectioon 2.2.

Network Management Knowledge	Network Knowledge
Manager	Configuration Model
Agent	Spatial Distribution Level
MIBs	Equipment Level
Alarm	Network Element Level
Diagnostic knowledge	Repetition Domain Level
Observation	Broadcast Domain Level
Symptom	Sub-network Domain Level
Hypothesis	Name Domain Level
Diagnostic	Application and Service Level
Network Performance Model	
Network Path Model	
Network Fault States Model	

Table 1 A part of a Network Management taxonomy focusing on the Diagnostic Task.

In the next sections we describe the two main domain models used in this work.

2.1 The Configuration Model

In order to reason about a fault in a communication network a system should have available the necessary knowledge about the network environment. Besides its availability, the knowledge must be well organized in such way that it becomes clear the relationship between components. In this work we propose a network knowledge organization in levels: the ***configuration levels***. The ***Configuration Model*** corresponds to the set of configuration levels, where each level contains important relationships between network terms. The levels described bellow correspond to the identified knowledge to be used for the diagnostic task:

Spatial distribution level: Describes how the equipments are spread over the buildings and rooms. This kind of information is important to diagnose building power-off or noise problems.

Equipment level: Describes the equipments, its ports and the connections among them. It also relates the ports, interface names and MAC address..

Network element level: In this level it is possible to represent the role of each equipment in the system. There are some equipments that have more than one role, for instance, a router with some ports configured as bridge and a computer with a hub board. In this case, it is necessary to decompose the equipment into other entities, called *Network Element*, each one with a clear role.

Repetition domain level: Allows to represent the repetition domains of the network and how they are interconnected. A repetition domain is a set of equipment ports to which an Ethernet packet is unconditionally repeated. This level is specially important for the diagnosis in order to identify what equipment interfaces are affected when there is an overload on a repetition domain.

Broadcast domain level: Represent the broadcast domains of the network and how they are interconnected. A broadcast domain is a set of equipment ports to which an Ethernet broadcast packet is unconditionally repeated. This level is important to identify what equipment interfaces may cause or be affected by a high level broadcast packets in a broadcast domain.

Subnetwork domain level: Represents the subnetworks and how they are interconnected. It also has information about the routing table of each equipment and IP-MAC addresses relationship. A subnetwork domain level is a set of IP interfaces that belong to the same subnetwork.

Name domain level: Describes how the names are assigned to IP addresses and how they are grouped into an Internet Domain Name.

Services and application level: This is a generic level and may be decomposed into several configuration modules. It is intended to model other necessary services and applications.

It is possible to view a generic TCP/IP local area network in those levels. In fact, the well known layers of the TCP/IP stack have driven us to organize the configuration model in the above levels. In the table below we show how the network representation levels can be related with the TCP/IP network layers.

The configuration levels described above identify the knowledge necessary for a range of diagnostic problems. Although the proposed model allows to represent a generic TCP/IP network environment this work initially focus on loss of communication caused by equipment problem or cable problem, diagnosis in a TCP/IP local area network. Therefore, it is only necessary to use knowledge about the Equipment Level, Network Element Level and Subnetwork Domain Level.

Configuration Model Level	TCP/IP layers
Application and Service Level	Application
Name Domain Level	Application
Network Domain Level	Internet
Broadcast Domain Level	Interface
Repetition Domain Level	Interface
Subnetwork Element Level	Interface, Internet
Equipment Level	Interface
Spatial Level	---

Table 2 A mapping between the TCP/IP layers and the network models proposed in this work.

2.2 The Network Path Model

This model corresponds to the human manager common sense knowledge about communication faults or performance problems in the network. In particular, this model specifies how to analyze paths of communication between the manager and the network elements in the presence of faulty elements. Reasoning about faults involves an important matter: the assumption of single or multiple faults. Therefore the Network Path Model depends on which assumption is made about the network in terms of single or multiple faults. In this paper we will make the assumption that only a single fault occurs in one complete pooling cycle.A pooling cycle is a periodic sample usually performed by the network management platform which sweeps the network devices periodicly.

When a diagnostic system receives an alarm that a particular network element E is not responding, it can conclude either E or some element in the path between the network manager and E is faulty. We call such set of network elements a hypothesis, that is, a hypothesis is a set of network elements in which at least one of them is faulty. Further alarms about other devices must be correlated with the current hypothesis. Bellow we describe some correlation rules used in this work:

Rule 1: Observation confirms the hypotheses and reduce the hypotheses. The path between the manager and an observed device is faulty and there is an intersection with the current hypotheses. In this case the new hypotheses is the intersection path.

Rule 2: Observation confirms the hypotheses but do not reduce the hypotheses. The path between the manager and an observed device is faulty and the intersection with the current hypotheses is the hypotheses itself. In this case the hypotheses remains the same.

Rule 3: Observation does not confirm the hypotheses and shows a new possible fault. The path between the manager and an observed device is faulty and there is no intersection with the hypotheses. Under the assumption of single fault, this case correspond to independent faults which can have independent diagnostic, e.g., each hypothesis will contain only one fault. This is equivalent to make an assumption of a minimal number of faults: if two faults can explain all symptoms, we will assume that two faults exist.

Rule 5: A normal observation contradicts part the hypotheses. The path between the manager and an observed device is OK and there is an intersection with the current hipotheses. In this case the observation eliminates part of the hipotheses, e.g., the intersection path.

Rule 6: A normal observation contradicts part the hypotheses. The path between the manager and an observed device is OK and the hipotheses is the intersection itself. In this case the observation deletes the hipotheses and the original fault can be considered as intermittent.

The knowledge described above define a part of a theory about how to process the result of communication path which will be used by the Problem Solving Method specified in the next Section.

3 Network Discovery System

An automatic or semi-automatic discovery system of the network configuration model implemented making use of a Management Network Platform. In general a network management platform has a *discovery module* which is capable of construct part of the network configuration model but usually not with the all need information for a diagnosis system. Such discovery system frequently collects information about Subnetwork Domain Level, sometimes about equipment level (equipment and interconnections) but not about other levels described in Section 2.1. In this work we have developed a ***Network Discovery System*** that can collet such information.

The prototype system runs on the Management Network Platform. It collects network level information, previously discovered by the platform, and interacts with the agents in order to gather additional information need for the construction of the Configuration Model.

To discover the network a Network Platform usually make use of ICMP echo, SNMP ARP table from MIB-II [RFC1213], SNMP route table from MIB-II [RFC1213], DNS requests, Bridge MIB [RFC1493] and Repeater MIB [RFC1516].

In this work the Configuration Model will contain only the levels that matter to solve *loss of communication* problems of a standard TCP/IP local area network, that is: the Equipment Level, the Subnetwork Level and the Network Element Level.

The output of this system can be used to support the network administrator to correct configuration problems before they cause new problems in the network.

4. The Problem Solving Method

According with the Knowledge Engineering AI field a general diagnostic Problem Solving Methods [4] [5] decomposes the diagnosis task into three sub-tasks [2] [15], as followed (see Figure 1):

Symptom Detection: it detects symptoms starting from observations done on the device. A symptom correspond to some abnormal observation. In the Communication Fault Diagnostic System (CFDS) the symptoms correspond to all the alarms of LOS type detected by the manager (abnormal observation).

Hypothesis Generation: given a set of symptoms, this sub-task determines possible causes (diagnostic) that explain the presence of the symptoms. Using the Configuration Model the CFDS calculates the set of network paths used by the manager to communicate with each NE corresponding to the LOS alarms in the symptoms set. Those path correspond to the initial hipotheses set. For every two network paths (hipotheses), this task uses the Network Path Model to compare the observations whith the model and to generate new hypotheses.

Hypothesis Discrimination: it analyzes the set of hypothesis generated by the previous task in order to determine the most probable one that is the best explanation for the observations. For that, the task can request additional observation on the device in order to get more information. In the CFDS additional observation correspond to all the devices which have none alarm of LOS associated to it, which we call *normal observation.* For this normal observation set, the CFDS calculates the new network paths using the Configuration Model. Finally, the Network Path Model is used to compare the observations with the model in order to discriminate the hypotheses.

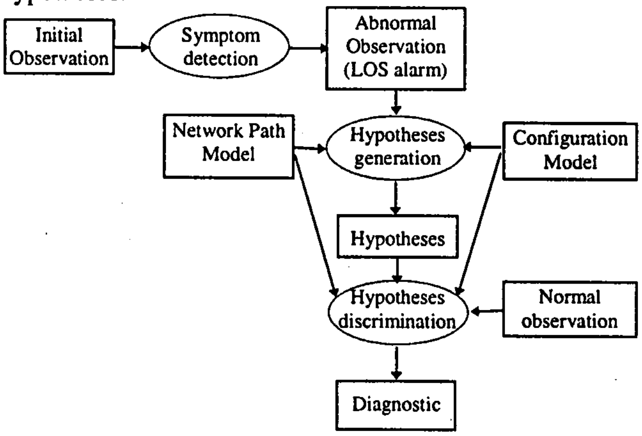

Figure 1 Data flow of the diagnostic method.

The calculus of the communication path: The two diagnostic subtasks: *Symptom Detection* and *Hypothesis Generation* have to execute the calculus of the communication path. This subtask starts from consulting the route table and identifying the routing elements sequence between the Network Management Platform and the network device in both directions. The system then identify the network elements between the routing elements, including also the connections between network elements.

5. Configuration Diagnosis System (CDS)

Because the Fault Diagnostic System could not reason about an inconsistent configuration model it was necessary to implement a system to analyze the discovered network configuration. Therefore, the Configuration Diagnosis System interacts with the *Network Discovery System* (see figure 2) in order to find out possible configuration errors. Only when no errors are found the Configuration Model become available to the Fault Diagnosis System. The CDS contains several implemented functions to check for the following characteristics: if all IP interfaces have assigned names; if there are no networks without any elements; if the network is connected, that is, if there is no connection information missing; if there are the same IP number assigned to different elements; if the routing tables are consistent with each other, etc.. There are others important checks to be made but those are the relevant ones to create a consistent configuration model.

6. Network Status Gathering System (NSGS) and Communication Fault Diagnosis System (CFDS)

The figure 2 presents the architecture of the prototype system. The Communication Fault Diagnosis System and the Configuration Diagnosis System are processes, whose programs are implemented in Prolog. The CFDS receives the Configuration Model from the Network Discovery System and receives asynchronously alarms related to *loss of communication* from the Network Management Platform.

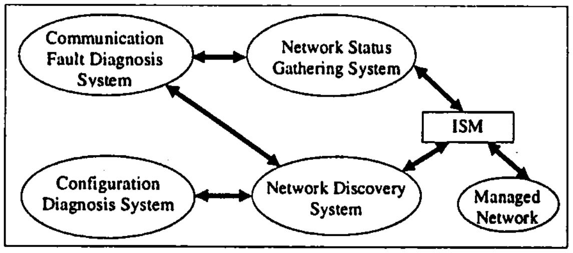
Communication Fault Diagnosis System
Network Status Gathering System
ISM
Configuration Diagnosis System
Network Discovery System
Managed Network

Figure 2 System Architecture

The Platform used is the *ISM OpenMaster* (*Integrated Management System* from Algar Bull Company), that follows the OSI Standard. There are two implemented applications running over the ISM Network Management Platform: the *Network Status Gathering* and the *Network Discovery* applications. The *IP Discovery* is a native ISM application used to discovery the equipments on the network. It collects some configuration data of each equipment and the subnetwork topology. This information is made available on *CMIS DB* service. The *Network Discovery* System access the *CMIS DB* and create the *Subnetwork Configuration Model.* For the construction of the *Equipment Level* and the *Network Element Level* models it was necessary to inquire agents about the Repeater MIB [RFC1515], the Bridge MIB [RFC1493] and others. This is done sending CMIP requests to the SNMP Agent Integrator. The SNMP Agent Integrator is an object manager that is responsible to converting CMIP requests from the Network Management Platform to SNMP requests.

The task of the *Information Gathering Application* is to receive all alarms and send only the relevant ones to the Diagnosis System in real time. In this case, the relevant alarms are those related to the communication state between the Network Management Platform and the equipment. The communication between the manager and an equipment may be in three states: *SNMP up*, *IP up* and *loss of signal*. The alarms are only generated on state transitions. Figure 3 presents the state diagram and the alarms that are generated when occur a transition. This mode of alarm generation characterizes the method presented in this paper as a *Passive by Transaction Diagnostic Method*. Different diagnostic methods can be specified according with the alarm generation mode as we will show in future publications.

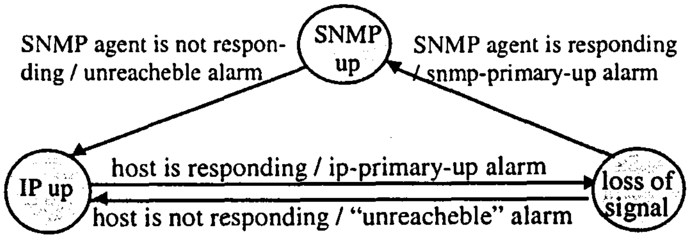
SNMP agent is not respon- ding / unreacheble alarm
SNMP up
SNMP agent is responding / snmp-primary-up alarm
IP up
host is responding / ip-primary-up alarm
loss of signal
host is not responding / “unreacheble” alarm

Figure 3 State Transitions Diagram of the communication between the manager and network devices.

The following table presents the alarms that are generated by the Agent Integrator Object Manager and that are used by the diagnosis system.

severity	alarm type	additional text
critical	communications	unreachable
major	communications	ip-primary-up
cleared	communications	snmp-primary-up

Conclusion

This paper presents a diagnostic reasoning approach based on a complete representation of the network, in its logical and physical levels, which we have called the network *Configuration Model*. To automatically construct the network configuration model, we developed a prototype of a subsystem for *Configuraration Dicovery*. In general, discovery systems in management plataforms only gather information at the level of networks. The discovery system we have presented uses information available in MIBs and can discover even physical connections between network elements.

We have developed two prototypes: a diagnostic system for communication faults and a diagnostic system for configuration error. The diagnostic of configuration error can detect and alert the network management team for configuration errors (inconsistencies between routing tables, for example) that can cause other communication faults or performance bottlenecks. The diagnostic of communication fault can determine a network element (of a minimal set of them) that are causing communication error, based on either the loss-of-signal alarms or on a specicically constructed query module that tests the acessibility of network elements periodically. Both subsystem were tested on articficially constructed networks.

It is important to point out that as part of the process of developing the prototypes we gained experience and insight on several issues that need to be addressed when developing diagnostics systems for networks, such as: 1) the specification of an ontology for the task of network diagnosis 2) identification of a taxonomy of network faults and abnormal behavior, since for each class of fault there is the need to develop a specific set of methods and models. 3) identification of the possible modes of interaction between the diagnostic system and the network. 4) specification of several diagnostic methods based on the different modes of interaction between the diagnostic system and the network. 5) specification of the many network models that are needed to represent the network. 6) modeling of the temporal aspects of network diagnosis.

References

[1] Abu-Hanna, A. Multiple Domain Models in Diagnostic Reasoning. Amsterdam, 1994. 169p. *Thesis (Ph.D.)*, University of Amsterdam.

[2] Benjamins, V. R. Problem Solving Methods for Diagnosis. Amsterdam, 1993. 172p. *Thesis (Ph.D.)*, University of Amsterdam.

[3] Breuker, J.; van de Velde, W.. CommonKADS *Library for Expertise Modelling Reusable problem solving components*. *IOS Press*, Amsterdam, 1994.

[4] Chandrasekaran, B. Generic Tasks in Knowledge-Based Reasoning: High-Level Building Blocks for Expert System Design. *IEEE Expert*, vol. 1, no. 3, p. 23-30, Fall 1986.

[5] De Kleer, J. and Willians B. C.. Diagnosing Multiple Faults. Artificial Intelligence 32 (1987) 97 - 130.

[6] Frohlich, P.; Jobmann, K.; Nejdl, W.; Wietgrefe, H. Model-based alarm correlation in cellular phone networks. In: *Fifth International Symposium on Modelling, Analysis and Simulation of Computers and Telecommunications Systems*. Proceedings, p. 197-204, 1997.

[7] Frontini, M.;Griffin, J.;Towers, S. A Knowledge-Based System for Fault Localization in Wide Area Networks. In: ISINM - *International Symposium on Integrated Network Management*, II, 1991. Proceedings, Elsevier, North-Holland, p. 519-530, 1991.

[8] Katzela, I; Schwartz, M. Schemes for Fault Identification in Communication Networks. IEEE/ACM Transactions on Networking, Dec, 1995.

[9] Kehl et al. Application of Model-Baesd Reasoning to the Maintenance of Telecommunication Networks. In: *5th IEA/AIE International Conference,*. Proceedings, Poderbon, Germany, June, 1992.

[10] Lemos, M. Um método de resolução de problema reusavel para diagnóstico automático no domínio de gerenciamento de falhas em redes de comunicação. *16o Simpósio Brasileiro de Redes de Computadores - SBRC98*. Page. 106-121. 1998

[11] Lewis, L. A case-based reasoning approach to the

resolution of faults in communication networks. In: ISINM - *International Symposium on Integrated Network Management.* Proceedings, Elsevier, North-Holland, 1993.

[12] Brugnoni et al. An Expert System for Real Time Fault Diagnosis of the Italian Telecommunications Network. In: *Third International Symposium on Integrated Network Management*, 18-23 April 1993 - 617-628.

[13] McDermott, J. Preliminary steps toward a taxonomy of problem-solving methods. In: Marcus, S. (editor), *Automating Knowledge Acquisition for Expert Systems*, p. 225-255. Boston, Kluwer, 1988.

[14] Nunes, C. M. Um Discriminador Inteligente de Eventos de Rede para o ambiente CINEMA. Porto Alegre, 1997, 143p. *Dissertação (Mestrado)* - CPGCC, UFRGS.

[15] Davis & Hamscher. Model-based reasoning: Troubleshooting. In Shrobe, H. E., editor. *Exploring Artificial Intelligence*, pages 297-346. San Mateo, California, Morgan Kaufmann.