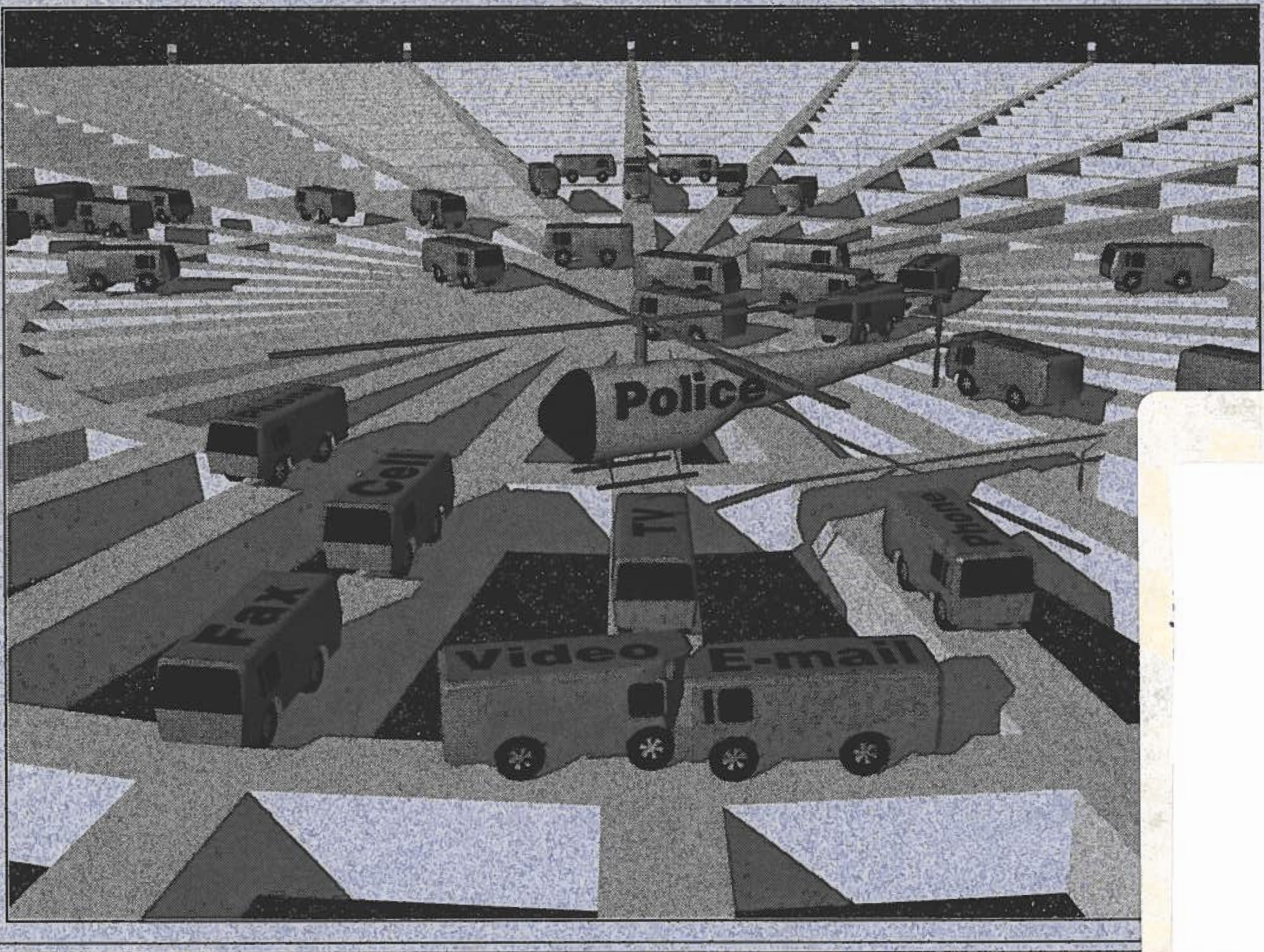
Canadian Artificial Intelligence
Intelligence Artificielle au Canada

An official publication of CSCSI, the Canadian Society for Computational Studies of Intelligence
Une publication officielle de la SCEIO, la Société canadienne pour l'étude de l'intelligence par ordinateur

Special Issue
Artificial Intelligence in Network Management

Edition speciale
L'intelligence Artificielle et Gestion de Réseau



Cover graphic design by Paul Amirault

Canadian Artificial Intelligence

Intelligence Artificielle au Canada

Spring/Summer 1997 | **No. 41** | **printemps/été 1997**

Canadian Artificial Intelligence is published three times a year by the Canadian Society for Computational Studies of Intelligence (CSCSI). *Intelligence Artificielle au Canada* est publiée trimestriellement par la Société canadienne pour l'étude de l'intelligence par ordinateur (SCEIO). Canadian Publications Mail Product Sales Agreement No. 507032.

ISSN 0823-9339

Contents / Contenu

Canadian Artificial Intelligence welcomes submissions on any matter related to artificial intelligence.

Please send your contribution, electronic preferred, with an abstract and a short bio to:
Sue Abu-Hakima, Institute for Information Technology
Editor, *Canadian Artificial Intelligence*
Institute for Information Technology
National Research Council Canada
M-50 Montreal Road
Ottawa, Ontario, Canada
K1A 0R6 or — suhayya@ai.iit.nrc.ca
Telephone (613) 993-8564 Fax (613) 952-7151

Book reviews and candidate books to review should be sent to:
Graeme Hirst
Canadian Artificial Intelligence
Department of Computer Science
University of Toronto
Toronto, Ontario, Canada
M5S 1A4 or — gh@ai.toronto.edu

Recycled / Recyclable

Suhayya Abu-Hakima

Special Issue: Artificial Intelligence in Network Management

Welcome to another special issue of *Canadian Artificial Intelligence*. This marks my first issue as sole editor. My co-editor Peter Turney has decided to step down after two years of excellent work. I would like to thank Peter Turney on behalf of the executive for the superb job he did editing *Canadian Artificial Intelligence*. Peter has to be commended for bringing the magazine to the World Wide Web in an electronic format to better prepare us for the next millenium.

I am very excited to bring you this special issue on AI in Network Management. Networking has fast become a cornerstone of our computing and communications needs. As networks such as telephone, wireless, cable, Internet, and Intranets converge, they become very complex to design, control, and manage and as such require the application of AI so that they may be more tractable. Diagnosis, machine learning, qualitative reasoning, and multi-agent systems are all branches of AI that have been applied to network management. Three articles on network management appear is this issue. The first is of a general nature and applies agents to the problem. The second addresses the use of swarm intelligence inspired by cooperative problem solving by ants in network routing problems. The third describes the use of constraint propagation for testing emulated local area networks over asynchronous transfer mode (ATM) networks. Also in this issue is a report on the successful 1996 International Workshop on Diagnosis DX-96 that was held in Val Morin, Quebec.

Our next issue will be a special issue edited by André Vellino of Nortel Technologies on AI and Logic, where are we at? The third issue this year will be a special issue on robotics and vision guest edited by Ramiro Liscano of NRC. I would encourage you to email me at suhayya@ai.iit.nrc.ca if you wish to contact the guest editors or to give me feedback on the magazine.

__Sue Abu-Hakima__ is the Group Leader of the Seamless Personal Information Networking (SPIN) Group at the Institute of Information Technology at NRC. SPIN focuses on the research and development of advanced software including agent and diagnostic technologies to real-world problems in telecommunications. She has been a researcher at NRC since 1987. She had previously worked at BNR starting in 1982. She has Doctorate and Masters degrees in AI from Carleton University's Department of Systems and Computer Engineering. She also has a B. Eng from McGill University's Department of Electrical Engineering with specialties in Communications and Computers.

On the Move

Effective July 1, 1997 the CSCSI web page has been relocated.
The new URL is http://www.cscsi.sfu.ca

Canadian Artificial Intelligence is accessible on our website to CSCSI/SCEIO members. If you would like to try electronic access, please send an e-mail message to peter@ai.iit.nrc.ca, with the following information:

- your full name
- your e-mail address
- a short user name (5-8 characters, for web access)
- a password (5-8 characters, for web access), and
- your CSCSI/SCEIO membership identification number, located on the mailing label of the magazine.

Artificial Intelligence Solutions for Network Management Problems

Babak Esfandiari

Résumé

Les normes de gestion de réseau ISO et UIT introduisent une représentation orientée-objet et un langage de communication de haut niveau pour les entités de gestion et les entités gérées dans un réseau. Nous voulons présenter dans cet article l'application de techniques de l'IA, par-dessus ces normes, pour résoudre certains problèmes communs de gestion et de supervision de réseau, tels que le filtrage d'alarme, le diagnostic automatisé et certains aspects en temps réel. Le filtrage d'alarme est un dispositif par lequel des rapports d'évènements non-voulus sont cachés à l'opérateur pour lui permettre de se concentrer sur l'information plus importante. Le diagnostic automatisé peuts' effectuer par la détection de séquences connues de rapports d'évènements.

De fait, certaines plates-formes actuelles de gestion de réseau offrent à l'opérateur des outils puissants, tels que des langages de programmation, pour l'aider à optimiser et à automatiser certaines tâches. Mais ces outils sont souvent difficiles à comprendre et par conséquent sous-utilisés. On a tenté d'utiliser des systèmes experts [Gaiti92] pour pallier à cela, mais le problème alors est de capturer et de formaliser l'expertise et de la faire évoluer avec le temps. Les règles des systèmes experts doivent aussi tenir compte de certains aspects des réseaux, comme l'incertitude sur la date de réception d'un évènement. Après une brève introduction de quelques concepts et normes de gestion de réseau, on suggère dans cet article d'utiliser des algorithmes de reconnaissance de scénario, l'apprentissage-machine et des techniques d'agent pour aider l'opérateur supervisant le réseau, et on montre comment on peut le faire en utilisant les normes.

1. Introduction

ISO and UIT network management standards introduce object-oriented representation and a high-level communication language for managing and managed entities in the network. The purpose of this paper is to present the application of AI techniques on top of those standards to address common network management and supervision issues such as alarm-filtering, automated diagnosis, and real-time aspects. Alarm-filtering is a feature that hides undesired event reports to the operator, to help him focus on more important information. Automated diagnosis can be achieved by the detection of known sequences of event-reports.

As a matter of fact, some current network management platforms offer powerful tools, such as programming languages, to the operator to help him optimize and automate some tasks. But those tools are often hard to understand and as a consequence they are underused. Some expert systems [Gaiti92] have been used to partly solve this, but then the problem is to capture and formalize the expertise, and to make it evolve as time goes by. The expert rules also have to take into account network related aspects such as the uncertainty of the date of the reception of an event.

After a brief presentation of some network management concepts and standards, this paper suggests the use of scenario recognition algorithms, machine learning, and agent-oriented techniques to help the network supervision operator, and shows how they can be built using the standards.

2. Presentation of Some Network Management Concepts and Standards

2.1 A Guideline for the Definition of Managed Objects (GDMO)

GDMO [Iso 90a] is an object-oriented formalism for the description of network resources. Each resource is considered a *managed object*. Objects having the same properties are considered instances of an object class, which belongs to an inheritance tree. An object class is a template, mainly defined by its:

- attributes
- actions: operations using the above attributes
- notifications: messages sent by the object to signal an event
- informal description of behaviour.

To enable querying upon a set of objects, the identifiers of the managed objects are defined within a *naming tree*. The hierarchy expresses that an object can contain, physically (components) or logically (directories, files, records, fields...), other objects, which can belong to different classes. The conceptual set of the objects of a managed system is called the Management Information Base (MIB).

2.2 The Common Management Information Service (CMIS)

CMIS [Iso90b] enables the dialogue between two network management entities by offering a set of primitives, which are used to send queries to a MIB or to report events. The main primitives are:

- M-GET: gets the attribute value of an object
- M-SET: modifies the attribute value of an object
- M-CREATE: creates an object with its default values
- M-DELETE: deletes an object
- M-ACTION: triggers an action on the named object
- M-EVENT-REPORT: reports an event.

The query primitives, besides the designation of the object, can have the following parameters, which allow the selection of a set of managed objects:

- The scope: determines the depth of the naming subtree on which the queries apply.
- The filter: selects a subset of the above subtree by setting conditions on attributes.

With those parameters, CMIS can be considered to be equivalent to SQL in term of expressiveness of queries, which is not the case of SNMP.

2.3 The Manager - Agent Model

In network management, the entity that sends queries and receives the answers and event notifications is called the *manager*. The managed entity, which receives queries, answers them, and sends event notifications, is called the *agent*. (Figure 1)

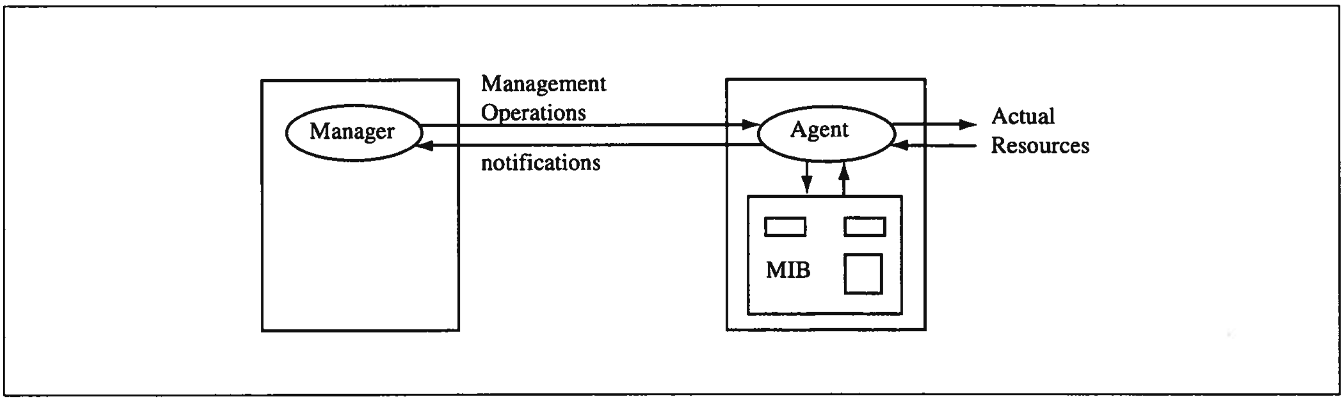


Figure 1. The Manager-Agent Model

3. Artificial Intelligence Techniques for Network Management

3.1 Scenario recognition

In network management, and more specifically in network supervision, there is a need to use a model to represent the managed system and changes occurring in it [Collectif95]. The system should be able to update its state after the reception of an M-EVENT-REPORT, so that possibly an action, such as the sending of a CMIS message or the filtering of the event, could be triggered. State machines of all kinds are the most simple formalism, consisting of oriented graphs with transitions designating the events required to move from one state to another. A path in the graph following the transitions is then called a *scenario.* A *scenario recognition* program can then take a decision when a scenario pattern has been completed. Such a program could therefore be used for automated fault diagnosis and alarm-filtering.

Often, more complex representation formalisms are needed, to avoid the need to describe all the states that a system can reach and to deal with temporal aspects. Temporal knowledge is required in network supervision for reasoning on events, actions, and change, in order to model facts such as: precedence, overlapping, simultaneity between events. While "numerical" approaches based on operations research are not adequate for symbolic reasoning, classical and modal logic approaches do not provide a good balance between expressiveness and algorithmic complexity.

The Chronicle model proposed by [Ghallab94] is based on two elementary types of formulae taken from the reified temporal logic: *events* and *holds.*

- a "hold" expresses that some ground domain attribute holds over some interval, for instance: hold (Trafic:normal, (t1,t3))
- an "event" specifies a discrete change of the value of an attribute, for instance: event (Transmission: (on, off), t2)

A chronicle model is a set of *event patterns* and temporal constraints between them and with respect to a context specified by *hold* assertions. If some observed events match the event patterns, and if their occurrence dates meet the specified constraints within its context, then an instance of this chronicle occurs, and an action specified in the chronicle is triggered. A *chronicle recognition system,* proposed by [Dousson96] processes the pattern matching in the following steps:

- Transform "holds" into "forbidden events," since an event should not change the value of the "held" attribute within the duration of the "hold."
- Create, if needed, a new instance of a possible chronicle and restrict the *window of relevance* (acceptable time

intervals for the expected events in order to complete a chronicle pattern) of all possible chronicles when a new event has been observed.

- Detect "deadlines" and occurrences of "forbidden events." In such cases, the corresponding chronicle will be removed from the list of the possible chronicles.
- Trigger off the action corresponding to the completed chronicle.

3.2 Interface Agents

As it was already stated in the introduction, having an expert system or a scenario recognition system is only a first step. The next problem that arises is the capture and formalization of the expertise. The recent concept of *interface agents* can be used in some extent to solve this problem. An interface agent is "a computer program that employs Artificial Intelligence techniques in order to provide assistance to a user dealing with a particular computer application. Such agents learn by 'watching over the shoulder' of the user and detecting patterns and regularities in the user's behaviour" [Maes94]. Since in network supervision one is interested in learning scenarios "on the fly," grammatical inference [Angluin82] is a possible choice to learn incrementally state-machines. [Schlimmer 93] describes an application of grammatical inference in a note-taking software.

However, in network supervision, the learning process must also take into account the fact that scenarios can overlap, as several faults can occur at the same time at different places in the network. We describe in [Esfandiari 96] an incremental chronicle learning algorithm that deals with overlapping. Here are, in short, the steps that the system follows:

- Chronicle *creation*: the system always stores the last received events in a buffer. When the operator triggers an event, then a chronicle is created, consisting in the set of the events, plus the time intervals between the events and the action.
- Chronicle *evaluation*: if the newly created chronicle leads to an action that is not the action field of a chronicle belonging to the *permanent chronicle base*, then it is compared to a possible chronicle with the same action field in the *temporary chronicle base*. A longest common subsequence calculus [Hunt77] is then performed between the event sequences of the two chronicles, and time constraints between the events are updated. The result replaces the "old" chronicle and a certain confidence value is then incremented for that chronicle.
- Chronicle *confirmation*: when the confidence value in a given chronicle reaches a certain predefined value, or when the operator decides that the chronicle is "mature" enough, then the chronicle is confirmed, i.e., transferred from the temporary chronicle base to the permanent one. It can then be considered as a new scenario to recognize by the chronicle recognition system.

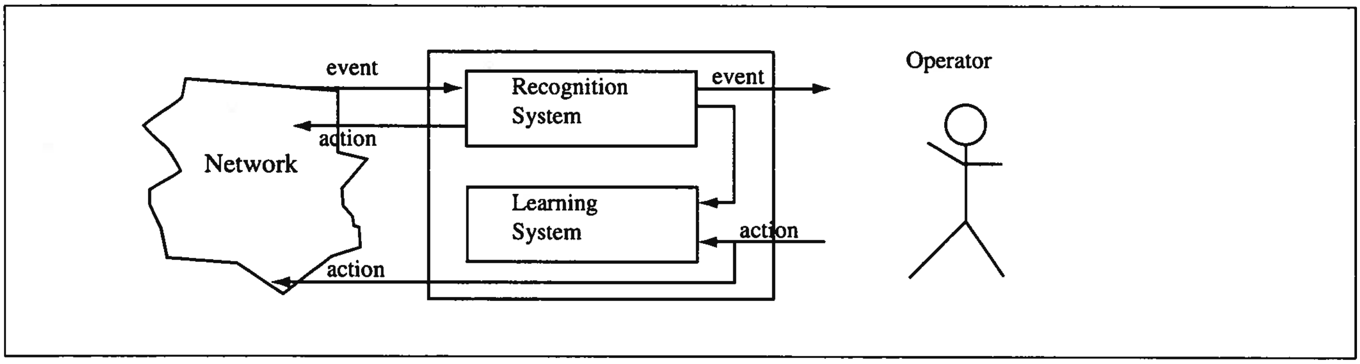


Figure 2. Interface Agents

3.3 Using CMIS in multi-agent protocols

We believe that CMIS primitives do not have to be exclusively used in network management. Even if they appear less expressive than KQML performatives [Genesereth 94], they have at least a very clear semantic (which is not always KQML's case, see [Cohen95]) and they proved to be sufficient in the *interaction protocols* [Demazeau 94] we implemented to enable the collaboration between several interface agents. Indeed, in order to increase the efficiency of our agent, we made it collaborate with other interface agents (also called *assistants*) through the network, to benefit from other operators experience. Thus, we involved the network management platforms in a true multi-agent system! Here are the protocols we used, inspired by [Lashkari94]:

- An *introduction protocol*, applied when inserting a new assistant in the network. To do so, the new assistant broadcasts its address to other assistants. Those assistants update their list of active assistants, and reply to it, giving it their address. Thus, the assistant will only send its requests to active assistants. In particular, when an assistant leaves the network, it only has to notify it to active assistants.
- An *exploration protocol*, a variety of the contract-net protocol, to find assistants dealing with similar patterns. This can be done by comparing confirmed chronicles that

trigger the same action from different chronicle bases. The result can be the attribution of a confidence rate to other assistants, depending on how close their chronicles are.

- A *query protocol,* to ask for some advice when facing a new case. When the confidence value of a chronicle reaches a certain *consultation threshold,* the assistant sends the sequence of events to other trusted assistants. The latter try to match the sequence with their own chronicles, and answer the possible behaviour they would have had in a similar situation.

All the above mentioned protocols used CMIS M-CREATE, M-GET, and M-ACTION primitives, making queries on dedicated GDMO classes such as Chronicle and Manager.

4. Conclusion

In this paper, we showed possible applications of artificial intelligence techniques in network management and supervision, using ISO network management standards. In fact, we believe that these kinds of applications underline the power of GDMO and CMIS as both simple and powerful knowledge modeling and querying languages, offering possibilities that simpler protocols such as SNMP do not offer.

The use of interface agents in network supervision can help the operator in using the maximum capabilities of the network management platform without having to use a programming language to customize the application.

Depending on the characteristics of the network, different formalisms can be used, ranging from simple state-machines to chronicle bases.

References

Angluin, D., *Inference of reversible languages*, In *Journal of the Association for Computing Machinery*, 29, p. 741-765, 1982

Cohen, P.R., Levesque, H.J., *Communicative actions for artificial agents*, In *ICMAS 95 Proceedings*, p. 65-72, 1995.

Collectif du projet inter-PRC, *Gestion de l'evolutif et de l'incertain dans une base de connaissances*, In *Actes des JN du PRC-GDR Intelligence Artificielle*, Nancy, 1995. Teknea.

Demazeau, Y., Boissier, O., Koning, J.-L., *Using interaction protocols to control vision systems*, In *International Conference on Systems, Man, and Cybernetics*, San Antonio, Oct. 94. IEEE.

Dousson, C., *Telecommunication Network: II- On-line chronicle recognition*, In Annals of Telecommunications, tome 51, no 9-10, p. 501-508, Sep. 96.

Esfandiari, B., Deflandre, G., Quinqueton, J., Dony, C., *Agent-oriented techniques for network supervision*, In *Annals of Telecommunications*, tome 51, no 9-10, p. 521-529, Sep. 96.

Gaiti, D., Pujolle, G., *L'intelligence dans les reseaux*, Eyrolles, Oct. 92.

Genesereth, M.R., Ketchpel, S.P., *Software agents*, In *Communications of the ACM*, 37(7):100-105, Jul 94.

Ghallab, M., *Past and future chronicles for supervision and planning*, In Jean-Paul Haton, editor, *Proceedings of the 14th Int. Avignon Conference*, p. 23-34, EC2, Paris, Jun. 94.

Hunt, J.W., Szymanski, T.G., *A fast algorithm for computing longest common subsequences*, In *Communications of the ACM*, 20, 5, p. 350-353, 1977.

ISO/DP 10165-4. *Information Technology - Structure of Management Information - Part 4. Guidelines for the Definition of Managed Objects*, Jun. 90.

ISO/DIS 9595. *Information Technology - Open Systems Interconnection - Common Management Information Service Definition*, Jan. 90.

Lashkari, Y., Metral, M., Maes, P., *Collaborative interface agents*, In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, AAAI-Press, 1994.

Maes, P., *Learning Interface Agents*, In *Proceedings of the 1994 International Symposium on Next Generation Computer Interface*, 1994.

Schlimmer, J.C., Hermens, L.A., *Software agents: Completing pattens and constructing user interfaces*, In Journal of Applied Intelligence Research, 1:61-89, 1993.

***Babak Esfandiari** received his PhD this year at the University of Montpellier in France, and has recently joined Mitel Corporation. His research areas include agent-based systems and network management.*

A Constraint Satisfaction Model for Testing Emulated LANs in ATM Networks

Mohammed H. Sqalli and Eugene C. Freuder

Résumé

Le mode de transfert asynchrone *(ATM – Asynchronous Transfer Mode)* est une technologie permettant d'intégrer tous les types de trafic dans un même réseau. L'émulation de réseau local *(LANE – Local Area Network Emulation)* est destinée à supporter les applications existantes de réseau local *(LAN – Local Area Network)* sur des réseaux ATM. Plusieurs compagnies produisent des systèmes de support de LANE sur ATM. On a donc besoin de tests d'interopérabilité et deconformité pour approuver ces systèmes. Une façon de tester l'interopérabilité des réseaux locaux émulés *(ELAN – Emulated LAN)* sur ATM est de surveiller le trafic entre le système du bout et le commutateur local et de détecter toute anomalie ou non-conformité de certains dispositifs aux spécifications du protocole. Dans cet article, nous modélisons les protocoles LANE en utilisant la représentation de satisfaction de contrainte. Les problèmes de satisfaction de contrainte *(CSP – Constraint Satisfaction Problems)* se sont montrés très utiles dans plusieurs applications dans le monde réel. Le test d'interopérabilité est effectué, avec des méthodes CSP, en associant les observations obtenues par surveillance et le modèle CSP des protocoles LANE.

Abstract

Asynchronous Transfer Mode (ATM) is a technology that allows the integration of all types of traffic in the same network. *LAN Emulation (LANE)* is destined to support the existing *LAN (Local Area Network)* applications over ATM networks. Many companies are manufacturing devices to support LANE over ATM. Hence, tests for interoperability and conformance are needed to approve these devices. One way to test the interoperability of the *ELANs (Emulated LANs)* over ATM is to monitor the traffic between the end system and the local switch, and detect any anomalies or non-conformance of certain devices to the protocol specifications. In this paper, we model the LANE protocols using constraint satisfaction representation. *Constraint Satisfaction Problems (CSP)* have proven very useful in many real-world applications. The interoperability testing is performed using CSP methods by associating the observations obtained from monitoring and the CSP model of the LANE protocols.

Introduction

Overview

Interoperability testing in networks is used to ensure that a device does what it is intended for. It is meant to supplement conformance testing by verifying that the end-to-end behavior of devices is compatible with the protocol specifications. Many formalisms and models have been used to perform this task, such as finite state machines (FSM).

A constraint satisfaction formalism has also been used as an extension to FSMs. In this paper, we propose to use CSP as a complete and direct model for the testing of ELANs. It is complete in terms of representation since no other formalism is needed to be used in conjunction with CSP, which was not the case before. Also, it is complete because it provides both the model of representation and the methods for problem solving. And, it is direct because it does not need intermediate representations (e.g., Extended FSMs) between the protocol specifications and the CSP model.

The first step consists of modeling the protocol specifications in a CSP representation. Observations are obtained by monitoring the traffic between the device to be tested and the network. Then, CSP methods are used to perform interoperability testing by tracking whether the observations satisfy all the constraints in the CSP graph (representation). The components of this model are presented in Figure 1. We also present a different way of solving the problem by using a partial constraint graph, focusing on a portion of the representation, to detect earlier if the device is not interoperable.

The CSP model of the initialization phase of the LANE protocol is presented to show how easy and powerful the CSP model can be in LANE, and that it can be extended to other domains.

Related Work

The trends of AI techniques applied to network monitoring and diagnosis are presented in [4].

Model-based approaches have been used in [5] for interpreting observations and diagnosis. The model presented in [5], called the system description SD, includes (possibly extended) finite-state machine (FSM) rules or constraints modeling agent communication behavior. In [6], a protocol is represented as a set of constraints derived from an Extended FSM (EFSM). It is stated in the same paper that several existing approaches to protocol diagnosis and testing are characterized in terms of the EFSM and the CSP formulation. Diagnosis can be viewed as a Partial Constraint Satisfaction Problem (PCSP) [7]. And, model-based diagnosis can be viewed as a constraint optimization problem [7]. The same technique is applied and extended to computer network software [8], where diagnosis is considered a Dynamic Partial Constraint Satisfaction Problem (DPCSP). The finite-state machine specification of a protocol is translated to a standard CSP representation, and configuration tasks are modeled as dynamic CSPs (DCSPs) [9].

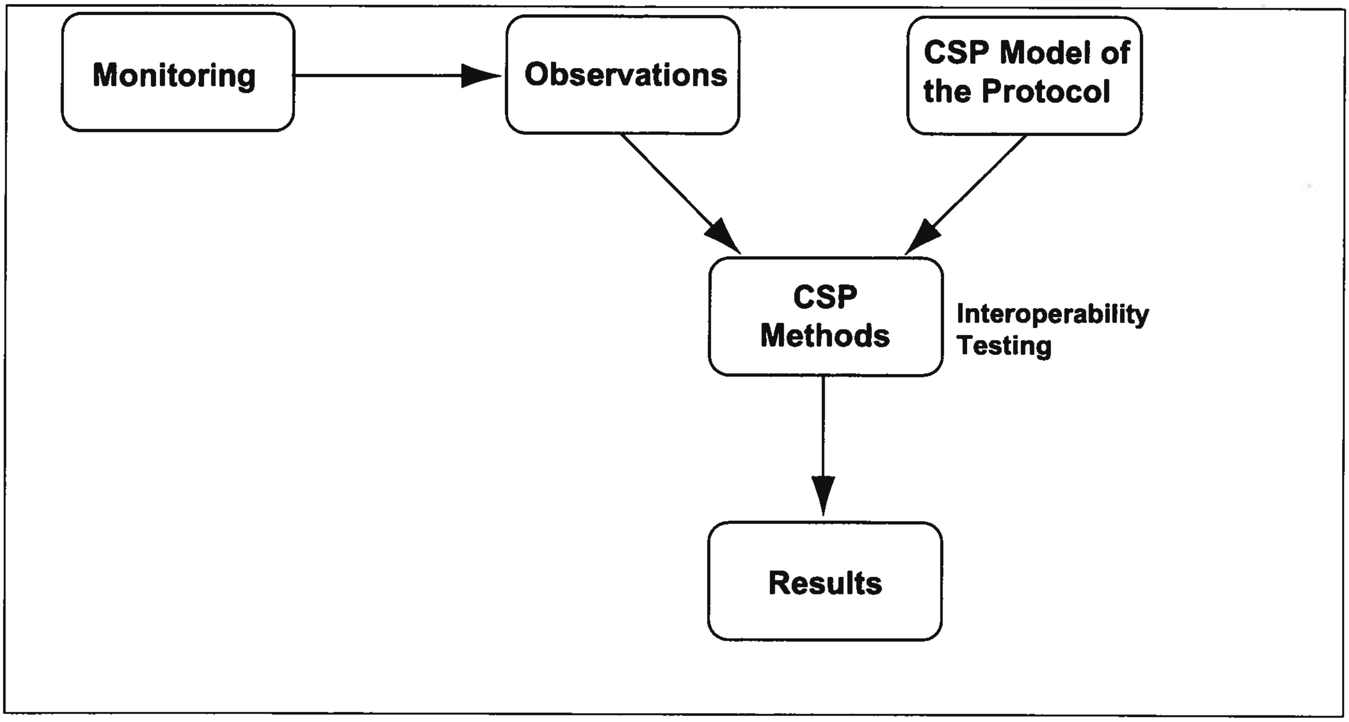


Figure 1. CSP Model Components for LANE Protocol.

In all these contributions, FSM is still used to represent the protocol specification. The CSP techniques applied are an extension to this approach, which may carry some disadvantages. Our goal is to model the protocol specifications as CSP in a natural way, which represents a new contribution in this field added to what has been done. The main contribution in this paper is that the CSP formulation of a protocol is derived from the specifications rather than from other formalisms.

LAN Emulation

ATM Networks

Asynchronous Transfer Mode (ATM) has emerged as a technology capable of supporting all classes of traffic (e.g., voice, video, data). ATM uses fixed-size cells, each having a 5 byte header and 48 byte payload. This allows the switching and multiplexing function to be done quickly and easily. ATM is a connection-oriented technology. Thus, for two end systems to communicate, they need to establish a fixed path through which they will send their data. Each connection is called a virtual channel (VC). The virtual path identifier (VPI) and the virtual channel identifier (VCI) are associated with a particular channel. Every cell will have this information (VPI and VCI) in the header. In ATM, the network can guarantee certain quality of service (QoS) requested by the user.

LANE Concept and Components

Before ATM fulfills its promise as a cost-effective technology for supporting future broadband multimedia services, the huge legacy of existing LAN applications needs to be readily migrated to the ATM environment. The LAN Emulation (LANE) specification defines how an ATM network can emulate a sufficient set of the medium access control (MAC) services of existing LAN technology (e.g., Ethernet and Token Ring), so that higher layer protocols can be used without modification. [10]

LANE consists of four major components:

- LAN Emulation client (LEC) – it is generally located in ATM end systems and serves as a proxy for LAN systems. It performs data forwarding, address resolution, and other control functions.
- LAN Emulation server (LES) – its main function is the support of the LAN Emulation address resolution protocol (LE-ARP), needed by a source LEC to determine the ATM address of the target LEC responsible for a certain destination MAC address.
- Broadcast/Unknown Server (BUS) – its task is to forward all multicast traffic to all attached LECs of the same ELAN. It handles data sent by an LEC to the broadcast MAC address 'FFFFFFFFFFFF', all multicast traffic, and initial unicast frames sent by a LEC before making a connection to a target LEC.
- LAN Emulation Configuration Server (LECS) – It assigns individual LANE clients to particular Emulated LANs (ELANs) by directing them to the LES that correspond to the ELAN. There is logically one LECS per administrative domain.

LANE Operation

The operation of a LANE system consists of an initialization/ configuration phase where the LEC must first obtain the ATM addresses of the LES and the BUS. For this, the LEC sets up a configuration-direct connection to the LECS (Figure

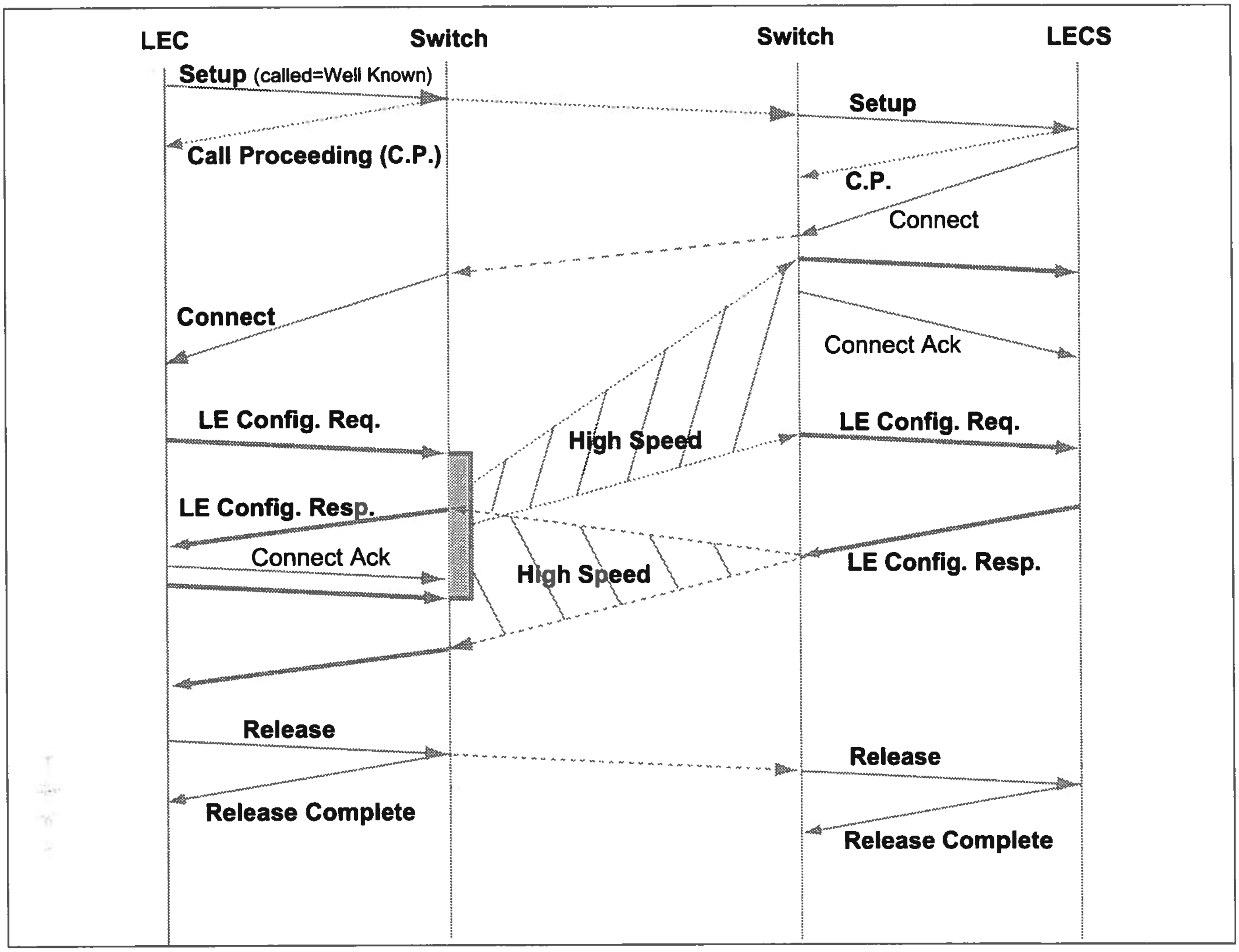


Figure 2. LEC Connection to the LECS (Phase 1).

2) which has a well known address. Once connected, a configuration protocol is used by the LECS to provide the LEC with the information it needs to connect into its target ELAN. This information includes the ATM address of the corresponding LES.

Once the LEC obtains the LES address, it may optionally clear the configuration-direct VCC (Virtual Channel Connection) to the LECS; then it sets up the control-direct VCC to the LES. The LEC then registers its own MAC and ATM addresses, and eventually all the MAC addresses it will represent as a proxy, with the LES. This is called the registration phase. At this stage, the LEC is part of the ELAN. The LES may open a point-to-multipoint connection to all LECs, called *Control Distribute Connection*. Then the LEC sends an LE-ARP request to the LES asking for the ATM address of the BUS. When the BUS address is received by the LEC, it sets up a connection to the BUS. More details can be found in [1], [2], and [10].

At this point, the LEC is ready to make connections and send/receive data to/from other LECs, in addition to multicast/broadcast, in the same ELAN. This is called *the Data Transfer Phase.*

Monitoring and Testing ELANs

We want to analyze all the traffic between a LEC and the local switch connected to it by placing a monitor between the two. The monitor gets all the data (observations) necessary to test the interoperability of the devices hooked to it. An observation is the data representing an event that occurred.

After we get the results of monitoring all the traffic between an end device (LEC) and the switch, we want to analyze the data obtained and determine if the device (LEC) is interoperable. One way to do it is by using finite state machines. An observation results in a transition from one state to another. If these transitions lead to a final state, then the device is interoperable. Otherwise, the device is not interoperable, and an analysis of the result is necessary to determine the cause of the problem.

But, finite state machines have a limitation when many alternatives are allowed, or when events may happen at anytime. For example, an event E1 may be received either before or after the event E2, or at anytime within future events. If we want to represent all the different combinations that might happen, we may have to come up with a very complex state machine. In some cases, we may have to represent the events using multiple state machines, then merge them. However, this does not solve the problem of complexity.

We propose to use the CSP formalism to analyze the data and test the interoperability of the device. CSP provides more flexibility in the representation of the events and constraints that must be satisfied.

CSP Overview

Constraint satisfaction is a powerful and extensively used artificial intelligence paradigm [3]. *Constraint satisfaction problems (CSPs)* involve finding values for problem variables subject to restrictions on which combinations of values are acceptable. A CSP graph is a representation of the CSP where the vertices are variables of the problem, and the edges are constraints between variables. Each variable has labels which are the potential values it can be assigned. CSPs are solved using search (e.g., backtrack) and inference (e.g., arc consistency) methods.

CSP representations and methods will be used for this particular problem since they provide a powerful tool in this case.

Statement of the Problem

Our goal is to test if a device is interoperable in an ELAN. We are interested in testing the behavior of an LEC that wants to connect to another LEC. In this paper, we model the LANE initialization phase (Phase 1) where the LEC makes a connection to the LECS, known as the configuration-direct connection (Figure 2). And, we are interested in the interoperability testing of this phase. The ELAN operation can be modeled in the same fashion as Phase 1, since the same aspects and challenges are encountered at both levels.

In Phase 1, as shown in Figure 2, the LEC sends a SETUP message to the LECS over the ATM network. The LECS address is well known by clients in the network. Then, the local switch may send back a CALL PROCEEDING message. When the LECS receives the SETUP message, it may send a CALL PROCEEDING message to the local switch. The LECS sends a CONNECT message to the LEC where a VCC is specified to be used for sending data between them. When the LEC receives the CONNECT message, it sends an LE CONFIGURATION REQUEST to the LECS. Then, the LECS sends back an LE CONFIGURATION RESPONSE with information including the LES address. In the meantime, the LEC sends a CONNECT ACK message to the local switch. The switch at the other end does the same with the LECS. The CONNECT ACK message may be received by the LECS either before or after the LE CONFIGURATION REQUEST. A RELEASE and a RELEASE COMPLETE message are sent to clear the connection between the LEC and the LECS.

Interoperability of a device is tested by looking at different parameters. Some of the data collected through monitoring represent these parameters. The first parameter is the time; for a device to be interoperable, events must happen in an expected sequence. The other parameters are either conditions that must be true, or actions that must be taken when an event occurs. Most of the actions are assignments of values to some variables, and most of the conditions are testing whether two variables are assigned the same value. Figure 3

Event	Source	Conditions	Actions
Setup	User	Call# = Well_known Address	ReqID = To_LECS_Call_Ref
Call Proceeding	Network	ReqID = To_LECS_Call_Ref	VCC = To_LECS_VC
Connect	Network	ReqID = To_LECS_Call_Ref	If (Call Proc. not received) VCC = To_LECS_VC
Connect ACK	User	ReqID = To_LECS_Call_Ref	
LE_Config Req.	User	VCC = To_LECS_VC	ReqID = Config_Req_ID
LE_Config Resp.	Network	VCC = To_LECS_VC ReqID = Config_Req_ID Resp_OK = T	LES_Addr = LES Address
Release	User	ReqID = To_LECS_Call_Ref	
Release Complete	Network	ReqID = To_LECS_Call_Ref	

Figure 3. Conditions and Actions in Phase 1.

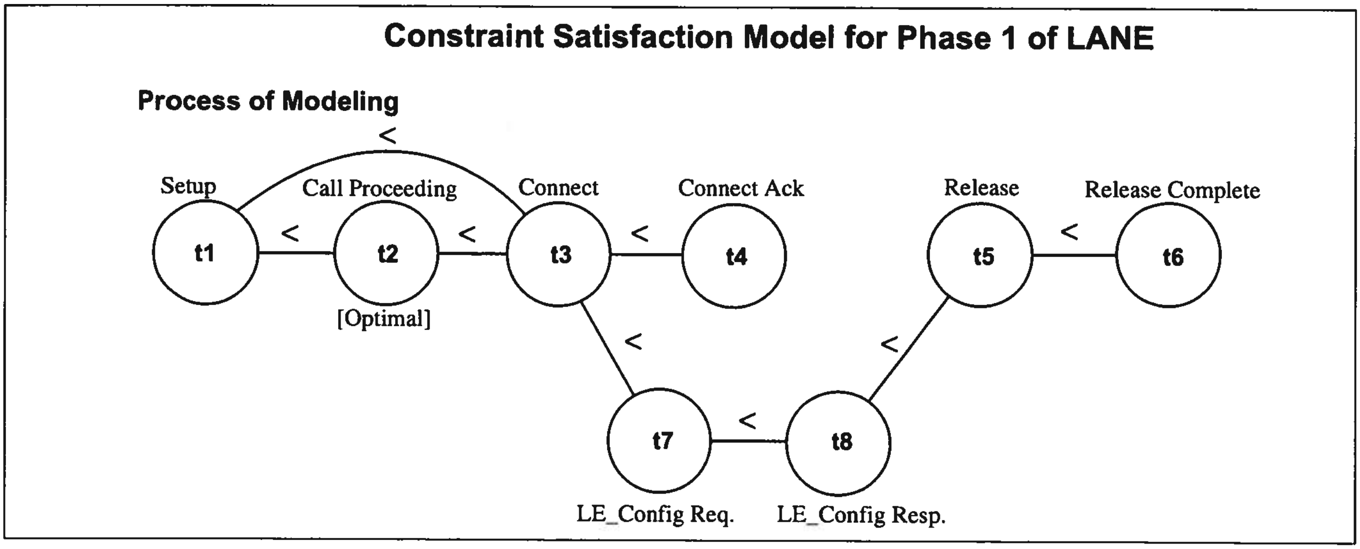


Figure 4. Basic CSP Graph (Time Constraints) of Phase 1.

shows these events with the parameters: source, conditions, and actions. For example, in the *setup event*, the *source* variable must be the user (U), the condition that must be true is that the call# must be the well known address. If so, the action taken is the assignment to the variable ReqID of the call_ref of the connection to the LECS. The ReqID variable will be used to test other future conditions. The timing sequence of the events is represented in Figure 4. For example, t8< t5 means that the LE_Config Response message must be received before the Release message.

In the state machine representation, the time parameter is an inner property. However, the conditions and actions (Figure 3) must be supported by other mechanisms (e.g. EFSM). The events that may happen at any time, while other events occur, add extra difficulty to the state machine. We call them *anytime events*. One example of these events taken from Phase 1 is the "CONNECT ACK" message (Figure 2). From the protocol specifications [2], this event may happen at anytime after the "CONNECT" event, either before or after some of the future events (LE_Config Request, ...), and it must occur only once. So, in terms of finite state machines, we may not be able to express that unless we do some extra work. We present in the next section a constraint satisfaction model that will solve most of the problems caused by these parameters.

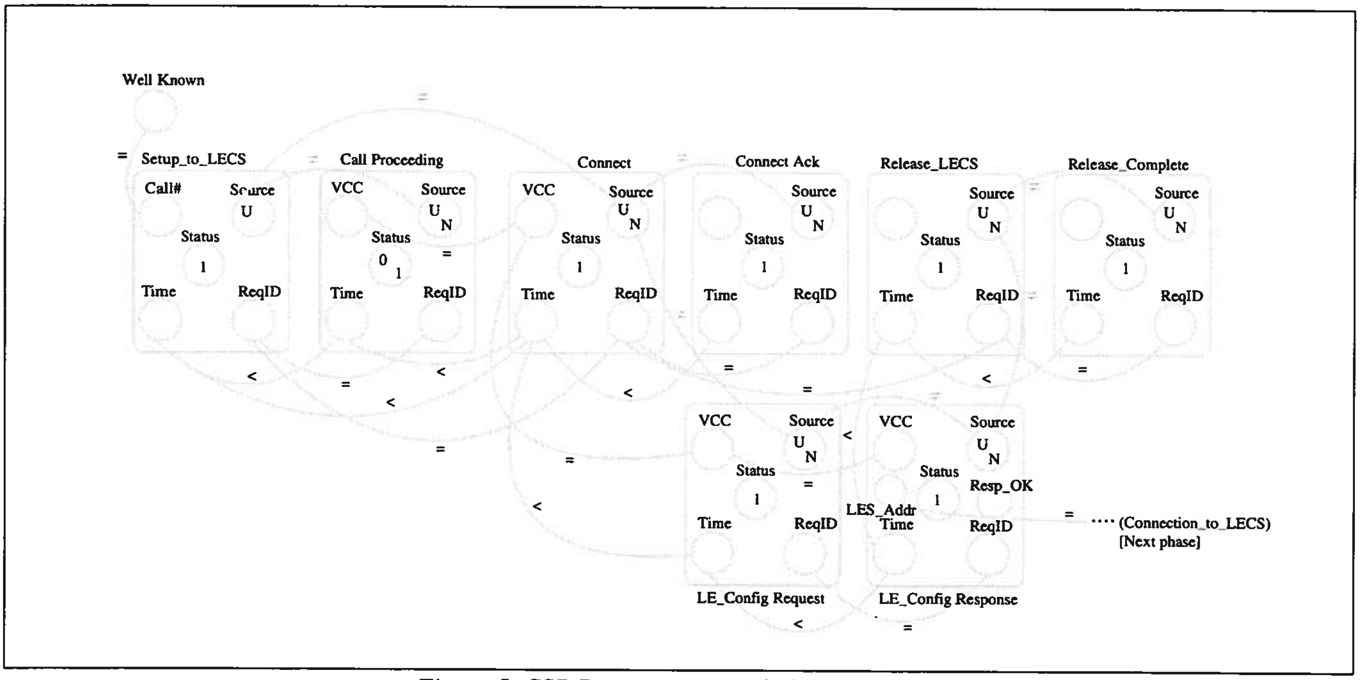


Figure 5. CSP Representation of Phase 1.

Constraint Satisfaction Model for Phase 1 of LANE Process of Modeling

We are interested in Phase 1 as described before, consisting of a LEC making a connection to the LECS.

The CSP representation we propose states that all the parameters will be represented as variables or constraints, thus eliminating the need for any extra work caused by the conditions, actions, and anytime events. Figure 4 shows the CSP representation of the *time* parameter of the Phase 1 events. The conditions and actions used with the state machine can be represented as constraints between variables (events' parameters), as shown in Figure 5. For example, the ReqID of the *Setup* message must be *equal to* the ReqID of the *Connect* message.

The steps we follow to model the testing of ELANs are:

1. Identify uniquely each event using the event name, the address, the timestamp, etc. In the case where we have more than one event with the same name (e.g., setup), the address can be used to identify them (i.e., LECS Address, LES Address, ...)
2. Represent the events into a constraint graph (Figure 5) where the variables are parameters (e.g., Address, Status, Time, ...) of the events (e.g., SETUP, CONNECT, ...), or other extra variables (e.g., well_known Address). The variable labels are the values that may be assigned to these variables (e.g., U or N for variable *source*), and the edges are the constraints we need to satisfy such as the order of receiving events, or the value a parameter may take. In the example, we have *equal to* constraint between the VCC variable of the Connect message and the VCC variable of the LE_Config Request.
3. Get the input data by monitoring the devices (Observations).
4. Use the event identifier (e.g., SETUP_TO_LECS) to map the events' parameters into variables, and assign values to them. We may wait until all the graph is instantiated (i.e., all events have occurred), then check whether all the constraints are satisfied or there is no violation. Alternatively, we may assign values to variables as events occur, and test if the LPCG is satisfied. A LPCG *(Logical Partial Constraint Graph)* is the portion of the constraint graph that has been so far instantiated. If there is a violation of a constraint, the process is stopped and a failure is reported. This allows us to detect errors earlier. Otherwise, the LPCG becomes larger by adding more variables to it until all the constraint graph is instantiated, which means that all the constraints are satisfied and a success is reported.
5. Test if all the constraints are satisfied after instantiating all the variables.
6. Report the results (OK or Error).

CSP Model

Figure 5 represents the CSP model of Phase 1. Each event has many parameters. The SETUP event has CALL#, SOURCE, TIME, REQUEST_ID, STATUS as parameters. Each of these parameters represents a vertex (variable) in the CSP graph. The labels in each vertex (values inside the vertex) are the values a variable may take, called the variable's domain. When no labels are present, the variable can be assigned any value. The constraints may be either unary, binary, ternary, etc. In Figure 5, there are unary and binary constraints. The unary constraints are the restrictions on the variable's domain. For example, in the Setup_to_LECS event, the variable 'status' can be assigned the value '1' only, meaning this value is mandatory; in Call_Proceeding event, since this value is optional, the associated variable may take either value 0 or 1. In the same way, the source can be either the user side (U) or the network side (N). The Well_known variable is assigned the value of the LECS address, which is known by all the clients on the network. The binary constraints are restrictions on the relation between two variables' domains. For example, there is a < constraint between the time variable of the Connect message and the time variable of the LE_Config Request.

CSP Methods and Problem Solving

Many CSP methods can be used to solve this problem, but mainly, there are two approaches widely used, either one of them or both: *Inference* and *Search*. After collecting all the observations from monitoring, we can assign the values to their corresponding variables (events' parameters). If all the constraints in the CSP graph are satisfied, the result of testing is positive; otherwise, we have to look for the constraints violation to know the cause of the error. This can be done after instantiating all the CSP graph. In this case, we test if all the constraints are satisfied. For every violated constraint found, an error is reported. This allows us to get all the violated constraints in the graph at the expense of waiting for all the events to occur to instantiate all the graph. However, this may be too costly when there are too many events, while the problem might be detected after the first few observations. To cope with this problem, we can use the LPCG by instantiating the events variables that have occurred so far, and as soon as we have a constraint violated, we announce that there is an error caused by the events received, and the process is stopped. This allows us to detect the first violation and which variables have caused it. Otherwise, we continue until all the graph is instantiated and no constraint is violated.

Advantages over Other Methods

This approach presents a novel contribution for testing protocols, which gets away from the use of FSMs with all the limitation they involve. This approach also gives a simple and natural way of modeling protocols, since we have to represent only what is stated in the specifications, without having to think about all the combinations that might happen with a given protocol. The advantages of this approach over other approaches are:

- We do not have to specify relations between events if they are not stated in the specifications. As we have seen before in Phase 1, the CONNECT_ACK may happen at anytime. Extra work is needed if FSM is used, and some knowledge of what may happen in all cases is needed. This is not the case with CSP models.
- The model is straightforward and simple, because it is derived from the protocol specifications, rather than from the interpretation of what may happen.
- The model is powerful, since with a simple representation we can get rid of extra work.
- The model is a natural way of representing what is expected to happen. All that is needed is translating the protocol specifications into a CSP representation.

Future Work

- We can improve the CSP representation by having many small subgraphs instead of one (Figure 5). We can obtain subgraphs like the one we used to represent the *time* constraint having all the *time* variables (Figure 4). We can have a subgraph for the *source* variables, another one for the *ReqID* variables, etc. In addition, some constraints link variables of different subgraphs. This representation is simpler to handle when the graph becomes very large.
- We need to implement and test the model that we described in this paper, and get the experimental results such that more situations can be analyzed.
- We need to extend this work, focused on Phase 1 only, to all the other phases.
- We need to extend this model to test other protocols.

Conclusion

This paper steps beyond the use of FSM in modeling and uses a pure CSP model for LANE over ATM network. We only presented the model of Phase 1 of the LANE protocol, which consists of the LEC making a connection to the LECS. The modeling of all the phases in ELAN operation can be done by extension of what is presented here. The use of constraint satisfaction models is a first step to solve many problems encountered in LANE testing in particular, and in protocol testing in general.

Acknowledgment

This material is based on work supported by the InterOperability Lab. at the University of New Hampshire, and by the National Science Foundation under Grant No. IRI-9504316. The first author is a Fulbright grantee sponsored by the Moroccan-American Commission for Education and Cultural Exchange. Special thanks to Ron Pashby for encouragement, discussions, comments, and his ideas on LANE operation and modeling. This work has benefited from discussions with Mihaela Sabin, Robert D. Russell, and Rakesh Thapar.

References

[1] Alles, A. 1995. ATM Internetworking. Cisco Systems, Inc. May 1995.

[2] The ATM Forum, Technical Committee. 1995. LAN Emulation Over ATM, Version 1.0. January 1995.

[3] Freuder, E., Mackworth A. 1992. Special Volume, Constraint-Based Reasoning, *Artificial Intelligence*, 1992, 58.

[4] Leckie, C. 1995. *Experience and Trends in AI for Network Monitoring and Diagnosis. Proceedings IJCAI-95 Workshop on AI in Distributed Information Networks.*

[5] Riese, M. 1993a. *Diagnosis of Communicating Systems: Dealing with Incompleteness and Uncertainty. Proceedings IJCAI-93*, 1480-1485.

[6] Riese, M. 1993b. *Diagnosis of Extended Finite Automata as a Constraint Satisfaction Problem. Proceedings DX-93, 60-73.*

[7] Sabin, D., Sabin, M., Russell, R., and Freuder, E. 1994. *A Constraint-Based Approach to Diagnosing Distributed Software Systems. Proceedings DX-94.*

[8] Sabin, D., Sabin, M., Russell, R., and Freuder, E. 1995. *A Constraint-Based Approach to Diagnosing Configuration Problems. Proceedings IJCAI-95 Workshop on AI in Distributed Information Networks.*

[9] Sabin, D., Sabin, M., Russell, R., and Freuder, E. 1995. *A Constraint-Based Approach to Diagnosing Software Problems in Computer Networks. Proceedings CP-95.*

[10] Siu, K., and Jain, R. *A Brief Overview of ATM: Protocol Layers, LAN Emulation, and Traffic Management. ACM SIGCOMM, Computer Communication Review.*

***Mohammed H. Sqalli** is a Ph.D. student in Engineering (Systems Design) at the University of New Hampshire and a member of its Constraint Computation Center. He received his M.S. in Computer Science from the University of New Hampshire in 1996. He is a recipient of a Fulbright Scholarship, and the second prize student paper award at Diagnosis: DX-96. His home page is: http://www.cs.unh.edu/~msqalli.*

***Eugene C. Freuder** is a professor in the University of New Hampshire Department of Computer Science and Director of its Constraint Computation Center. He is a Fellow of the American Association for Artificial Intelligence. He is the founding editor of Constraints, An International Journal (Kluwer Academic Publishers) and a member of the Organizing Committee of the International Conference on Principles and Practice of Constraint Programming. His home page address is: http://www.cs.unh.edu/ecf.html.*

Swarm Intelligence and Problem Solving in Telecommunications

Tony White

Résumé

Cet article décrit comment des agents inspirés de la biologie peuvent être utili séspour résoudre des problèmes de contrôle en télécommunications. Ces agents, inspirés par le comportement fureteur des fourmis, affichent les qualités désirables de simplicité d'action et d'interaction. La collection d'agents, ou système-essaim, ne s'occupe que de l'information locale et démontre une forme de contrôle distribué, les communcations entre agents étant effectuées à travers l'environnement. Dans cet article, nous explorons l'application d'agents-fourmis au problème d'acheminement dans les réseaux de télécommunications.

Abstract

This paper describes how biologically-inspired agents can be used to solve control problems in Telecommunications. These agents, inspired by the foraging behaviour of ants, exhibit the desirable characteristics of simplicity of action and interaction. The collection of agents, or swarm system, deals only with local knowledge and exhibits a form of distributed control with agent communication effected through the environment. In this paper we explore the application of ant-like agents to the problem of routing in telecommunication networks.

Introduction

The notion of complex collective behaviour emerging from the behaviour of many simple agents and their interactions is central to the ideas of Artificial Life. Nature provides us with many examples of social systems where individuals possess simple capabilities when compared to their collective behaviours which are much more complex. Such systems span several levels of evolutionary complexity, from simple bacteria [Shapiro 88], to ants [Goss et al. 90], [Franks 89], caterpillars [Fitzgerald and Peterson 88] and beyond.

The continuing investigation and research of naturally-occurring social systems offers the prospect of creating artificial systems that are controlled by emergent behaviour and promises to generate approaches to distributed systems management found, for example, in telecommunications networks.

Controlling distributed systems such as those found in telecommunications networks by means of a single central controller, or requiring each controlling entity to have a global view of the system, has many disadvantages. In the case of the single controller, a considerable quantity of information must be communicated from the network to the controller, necessitating the sending of data from all parts of the network to the centralized control point. These systems scale badly due to the rapid increase in the amount of data that must be transferred and processed as the network increases in size. Providing a single point of control also provides for a single point of failure, a highly undesirable characteristic of any system. In the case where multiple global views are constructed and maintained, the problem of synchronization of such views can lead to instability and excessive use of communications capacity. The optimal design of a centralized controller is often difficult to achieve in that design decisions must be made based upon a static (and idealized) view of the way in which demands on resources in the network are likely to change.

Decentralized control mechanisms need not suffer from the above problems and potentially can take advantage of local knowledge for improved use of network resources.

In this paper we describe the essential principles of Swarm Intelligence (SI) and how an understanding of the foraging behaviours of ants [Beckers et al. 92] has led to new approaches of control in telecommunications networks.

Swarm Intelligence and the Ant Colony

Swarm Intelligence [Beni and Wang 89] is a property of systems of unintelligent agents exhibiting collectively intelligent behaviour. An agent in this definition represents an entity capable of sensing its environment and undertaking simple processing of environmental observations in order to perform an action chosen from those available to it. These actions include modification of the environment in which the agent operates. Intelligent behaviour frequently arises through indirect communication between the agents; this being the principle of stigmergy [Grassé 59]. It should be stressed, however, that the individual agents have no explicit problem solving knowledge and intelligent behaviour arises as a result of the actions of societies of such agents.

Individual ants are behaviourally simple insects with limited memory and exhibiting activity that has a random component. However, collectively, ants manage to perform several complicated tasks with a high degree of consistency. Examples of sophisticated, collective problem solving behaviour have been documented [Frank 89; Hölldobler and Wilson 94] including:

1. forming bridges;
2. nest building and maintenance;
3. cooperating in carrying large items;
4. finding the shortest routes from the nest to a food source;
5. regulating nest temperature within a one degree celcius range;
6. preferentially exploiting the richest source of food available.

In the above examples, two forms of stigmergy have been observed. Sematectonic stigmergy involves a change in the physical characteristics of the environment. Nest building is an example of this form of communication in that an ant observes a structure developing and adds its ball of mud to the top of it. The second form of stigmergy is sign-based. Here, something is deposited in the environment that makes no direct contribution to the task being undertaken but is used to influence the subsequent behaviour that is task related.

Sign-based stigmergy is highly developed in ants. Ants use highly volatile chemicals called pheromones (a hormone) to provide a sophisticated signalling system. Ants foraging for food lay down quantities of pheromone marking the path that it follows with a trail of the substance. An isolated ant moves essentially at random but an ant encountering a previously laid trail will detect it and decide to follow it with a high probability and thereby reinforce it with a further quantity of pheromone. The collective behaviour which emerges is a form of autocatalytic behaviour where the more the ants follow the trail the more likely they are to do so. The process is characterized by a positive feedback loop, where the probability that an ant chooses any given path increases with the number of ants choosing the path at previous times.

Routing

The motivation for exploiting the ant metaphor for routing in telecommunications networks arises from the fact that routing systems frequently depend upon global information for their efficient operation. Ant systems do not need such global information, relying instead upon pheromone traces that are laid down in the network as the ant, or agent, moves through the network. Global information is frequently out of date and transmission of the information required from one node to all others consumes considerable network bandwidth.

Ideally, we would like to have the network adapt routing patterns to take advantage of free resources and move existing traffic if possible. To date, two applications of the ant metaphor for routing have been documented [White 96], [Schoonderwoerd 96]. This paper describes the approach of [White 96].

Ant Routing

The "Routing by ants" system consists of three agent types; explorers, allocators, and deallocators. Explorer agents exhibit the foraging behaviour of ants and preferentially follow trails of pheromones laid down by previous explorers. Allocator agents traverse the path determined by explorer agents and allocate the bandwidth on the links used in the path. Similarly, when the path is no longer required, deallocator agents traverse the path and deallocate the bandwidth used on the links.

The system works in the following way. A call request arrives at a given node. The call is either a point to point (P2P) or point-to-multipoint (P2M) request. For P2P requests a new species of ant (agent) is created and sent out into the network. For a P2M request *n* agents of a new species are created and sent out into the network. These explorer agents execute the following algorithm:

```
1. Initialize
   set t:= 0
   For every edge (i,j) set an initial value T_ij(t)
   for trail intensity. Place m ants on the source node.
   [Generate new explorers at freq. e_f ]
2. Set s:= 1 { tabu list index)
   for k:= 1 to m do
       Place starting node of the kth ant in tabu_k(s).
3. Repeat until dest'n reached:
   Set s := s + 1
   for k:=1 to m do
   Choose node j to move to with prob p_ij^k (t)
   Move the kth ant to node j.
   Update explorer route cost: r_k = r_k + c(i,j)
   if (r_k > 'max)
       kill explorer_k
       Insert node j in tabu_k (s).
       At destination go to 4.
4. While s > 1
   traverse edge (i,j)
   T(i,j) = T(i,j) + p_e
   s := s - 1
5. At source node do:
   if (path_e = pathBuffer * d)
       create and send allocator agent
   if t > Tmax
       create and send allocator agent
```

Explorer agents are created at a given frequency e_f and continue to be created and explore during the lifetime of the call. In this way, it is possible to have recovery from node or link failure and (potentially) have the system re-route calls in order to overcome temporary congestion situations.

When explorer agents reach their destination, they backtrack along the route chosen and drop pheromone in order to mark the path. Upon arrival back at the source node, a decision is made whether or not to send an allocator agent. The decision is made based upon *m* previous allocator agents' paths. If *p%* of the agents follow the same path, the path is said to have emerged and an allocator agent is created and enters the network in order to allocate bandwidth. In the case of P2M allocator agents, the decision is made based upon whether the spanning trees chosen are the same. A convenient property of the P2M path search is that new connections can be added dynamically as remote sessions come online as would typically be the case in a distance learning application. Potentially the entire spanning tree found by the P2M agents might then change as a more efficient multi-cast solution is found.

Allocator agents traverse the path indicated by the highest concentrations of the pheromones dropped by their associated explorer agents. It is possible that network bandwidth has already been allocated by the time the allocator agent is sent and in this case the allocator agent backtracks to the source node rolling back resource allocation and decreases pheromone levels. A decision to re-send an allocator agent is made at a later time after a back-off period has been observed. During the back-off period explorer ants continue to search for routes.

The probability with which an explorer agent (k) chooses a node j to move to is given by:

$$p_{ij}k\ (t) = [T_{ij}(t)\ ]^{\alpha}[C(i,j)]^{\beta} / N_k$$

$$N_k = \Sigma_k \text{ in } (S\text{-Tabu}(k))\ [T_{ij}(t)\ ]^{\alpha}[C(i,j)]^{\beta}$$

where α and β are control constants and determine the sensitivity of the search to pheromone concentration and link cost respectively. N_k is a normalization term that makes p_{ij}^k (t) a true probability. Low values of α indicate that the search process is insensitive to pheromone concentration, whereas low values of β indicate that link cost is unimportant. The balancing of these two parameters strongly affects the efficiency of the search process.

As a result of the sensitivity of the search process to parameter settings, the search process was made adaptive. The values α and β were not fixed for the all ants but allowed to vary based upon the effectiveness of the search resulting from them. In order to achieve this, each ant was given an associated fitness value – the cost of the path found – and whenever new ants were created α and β were determined by proportional selection of two parent ants based upon fitness followed by crossover and mutation.

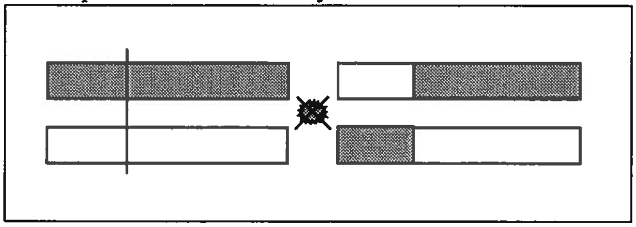


Figure 1. An example of crossover

The figure above shows the genotype of two ant parents that encode α and β. A single crossover point is chosen and two offspring are generated. One offspring is discarded and the other undergoes mutation as shown below.

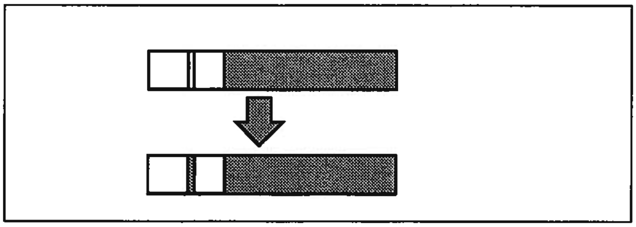


Figure 2. An example of mutation

As can be seen from the two figures, the resulting offspring is rarely the same as either of the parent. Using this approach significantly improved the search process as the agents learned efficient search parameters.

Summary

This paper has described a search process that solves the routing problem for networks containing both point-to-point and point-to-multipoint call requests. The process requires three agent types and is dynamic in nature, thereby allowing the potential for re-routing in situations where local congestion occurs. An interesting property of the process is the potential for reconfiguration of multi-cast path solutions as new sessions come online or existing sessions terminate. Results have shown that shortest path routes can be quickly computed and that response to failure events in the network is rapid.

Future Work

The system is currently being extended to allow for interactions between pheromone species. In the multi-pheromone ant colony system (mPAC), a chemistry $\mathbb{C}$, is defined for the system:

$$c_i: s_1S_1 + \ldots + s_mS_m \rightarrow s_1'S_1' \ldots + s_m'S_m'$$

The chemical reactions can be defined link by link, globally defined and several reactions are possible $\mathbb{C} = \{c_i\}$. Each reaction has an associated reaction rate (defined by Arrhenius equation), i.e., temperature dependent. It is envisaged that the addition of such a chemistry will provide for a mechanism to define behavior-based (subsumption) network management and control systems.

References

[Beckers et al 92] Beckers, R., Deneuborg J.L, and Goss, S., *Trails and U-turns in the Selection of a Path of the Ant Lasius Niger.* In *J. theor. Biol.* Vol. 159, pp. 397-415.

[Beni and Wang 89] Beni, G., and Wang, J., *Swarm Intelligence in Cellular Robotic Systems, Proceedings of the NATO Advanced Workshop on Robots and Biological Systems*, Il Ciocco, Tuscany, Italy, June 26-30, 1989.

[Franks 89] Franks, N.R., *Army Ants: A Collective Intelligence, Scientific American*, Vol 77, March-April, 1989.

[Fitzgerald and Peterson 88] Fitzgerald, T.D., Peterson, S.C., *Cooperative foraging and communication in caterpillars, Bioscience*, 38, 1988, pp. 20-25.

[Goss et al 90] Goss, S., Beckers, R., Deneubourg, J.L., Aron, S., Pasteels, J.M., *How Trail Laying and Trail Following Can Solve Foraging Problems for Ant Colonies*, in Hughes R.N. (ed.) NATO ASI Series, Vol. G 20, *Behavioural Mechanisms of Food Selection*, Springer Verlag, Berlin, 1990.

(continued on page 18)

Minimum Intelligent Signal Test: An Objective Turing Test

Chris McKinstry

Résumé

Un test de signal intelligent minimum (MIST - Minimum Intelligent Signal Test) est une forme de test de Turing conçu pour détecter l'intelligence de systèmes synthétiques et en fournir une mesure, requérant seulement de ces systèmes qu'ils répondent de façon binaire. Ce test peut être utilisé pour fournir une réponse automatisée à des systèmes génétiques, à des systèmes neuronaux ou à d'autres systèmes de corrélation statistique opérant dans le champ de la connaissance humaine.

Abstract

A Minimum Intelligent Signal Test (MIST) is a form of Turing Test designed to detect and provide a measure of intelligence for synthetic systems, while only requiring those systems to respond in a binary fashion. It may be used to provide automated feedback to genetic, neural, or other statistical correlation systems operating in the human knowledge domain.

Introduction

In 1950, Alan Turing proposed a test of machine intelligence now commonly referred to as the Turing Test (1). The premise of this test is rooted in the philosophy of other minds problem, which states that thinking can detect itself, but it can only infer the existence of thought in others through the actions of other thinkers. In other words, the appearance of thinking, is thinking. Turing predicted that by the year 2000, there would exist machines with the storage capacity of 10**9 bytes (2) that could fool the average questioner into thinking the machine was human for a period of 5 minutes, 70% of the time.

In 1991, Dr. Hugh Loebner, the National Science Foundation, and the Sloan Foundation started the annual Loebner Prize to offer $100,000 US to the creator of the first computer program to pass an unrestricted Turing Test (Epstein, 1992). At the end of 1995, hard disk drives with 10**9 bytes could be purchased for under $400 almost anywhere. Despite the prize and the availability of storage space, no computer program has yet been able to pass Turing's test, and the Loebner Prize remains unclaimed.

Minimum Intelligent Signal Test

Given a series of stimuli (items), a system being tested generates a binary response for each stimulus. Thus, a Minimum Intelligent Signal may be detected in the cumulative binary output of that system. A system that has a MIST score that is statistically different from a random system is said to be intelligent. A system that has a MIST score that does not differ statistically from the MIST score of an average human, is said to have the intelligence of an average human.

Method

1) N, items are generated: All items must be able to be responded to by systems (i.e., people) judged to have human intelligence (3), in a binary fashion. The response must have statistical stability, both when a human tries to give an intelligent response and when a human tries to evade (give a non-intelligent response).
2) Items are presented, and responses recorded: Items are presented in the highest common mode (4) between human and synthetic subjects. On subsequent re-trials, item order is re-randomized (5).
3) Double blind experimenter grades item/response pairs: For each item, judge the item/response pair either consistent or inconsistent with human intelligence. This grading procedure may be easily automated, reducing the chance of grading error or unforeseen bias.
4) Generate Score: Sum the total of the items judged consistent I (Intelligent), and sum the items judged inconsistent E (Evasive). Probability the system under consideration is intelligent and cooperative is $p(I)=I/N$. Probability system is intelligent and evasive is $p(E)=E/N$. Both probabilities must sum to 1.0.

Predictions

MIST's yield standardized probability scores. The judgment of which individual response to an item is intelligent has been made and validated prior to administration of the test, and is defined as being stable for normal human intelligence. Random systems which exhibit no intelligence will have MIST scores of p(I) and $p(E) = .5$, while intelligent systems (natural and synthetic) will have MIST scores of p(I) or p(E) that are statistically different from a random system.

Conclusion

Using a very large corpus of MIST item/response pairs, self organizing systems may develop human-like intelligence, with no programmer intervention. For example, a system could be developed that allows infinite input configurations and binary output, with some complex connectionist layer in-between. Using simple feedback, this type of system should automatically discover the correlation between input and output that we call human intelligence or common sense, that created the corpus. This process is analogous to building a medical CAT scan image; where many independent measurements from many points of view are statistically combined to form an image of the common cause of all the measurements.

Independent of its application as a feedback function for creating intelligent systems, MIST remains an objective method for describing emerging synthetic systems. The future may see MIST scores correlated with scores from more complex intelligence tests that yield a standardized IQ. This would allow synthetic intelligence to be expressed in the same manner as human intelligence, while only requiring the synthetic system to respond in binary, greatly simplifying development of such synthetic systems.

Authors Note

Currently, I am in the process of organizing an annual MIST contest modeled after the Loebner Prize, as well as creating a central MIST item repository (with a goal of 1 million validated items by January 1, 2000) for educational use. If you are interested in any aspect of the MIST contest or the repository project, please contact me directly or take a look at http://www.clickable.com/mist.html for current information.

References

(Turing, 1950) A. M. Turing, *"Computing Machinery and Intelligence," Mind,* Vol. 54, No.236, October 1950, pp.433-460

(Searle, 1980) J. R. Searle, *"Minds, brains and programs." Behavioral and Brain sciences,* 3: 417-424

(Epstein, 1992) R. Epstein, *"The Quest for the Thinking Computer," AAAI Magazine*, Vol. 13, No. 2, Summer 1992, pp. 80-95.

(Shieber, 1992) S. Shieber, *"Lesions from a Restricted Turing Test," Technical Report TR-19-92, Harvard University,* Sept. 1992, Revision 4.

Footnotes

(1) There have been numerous criticisms of Turing's test. (Searle, 1980; Shieber, 1992, et al.)

(2) Turing actually used 'numbers' and not the current term, 'bytes.'

(3) In practice, each item is defined as being stable for the subgroup of the population which they were validated against (i.e., undergraduate psychology students).

(4) The Loebner Prize has recently been modified to require all systems attempting to win the $100,000 prize accept audio/visual input in real-time as would a human. The author feels this is arbitrary and unfair to synthetic systems. Intelligence is not dependent on sight or sound, but rather on intelligent responses to various stimuli. With current systems, text would be considered the highest common mode.

(5) This precludes between item dependence; all items must be able to stand on their own.

***Chris McKinstry** is an Information Technology Consultant with LGS Group, Inc., one of Canada's largest information technology consulting firms. He has over ten years of software development experience, primarily in large database environments. Some of his clients have included Litton Systems Canada, Atomic Energy of Canada, Manitoba Public Insurance, and CP Rail. He may be reached via email at chris@clickable.com*

Swarm Intelligence and Problem Solving in Telecommunications

(con't. from page 16)

[Grassé 59] Grassé, P.P., *La reconstruction du nid et les coordinations inter-individuelles chez Bellicoitermes natalenis et Cubitermes sp. La theorie de la stigmergie: Essai d'interpretation des termites constructeurs.* In *Insect Societies*, Vol. 6, pp. 41-83.

[Hölldobler and Wilson 94] Hölldobler, B. and Wilson, E.O., *Journey to the Ants.* Bellknap Press/Harvard University Press, 1994.

[Shapiro 88] Shapiro, J. A., *Bacteria as multi cellular organisms, Scientific American*, 1988 pp. 82-89.

[White 96] White, A.R.P., *Routing with Ants*, Nortel internal report, 1996.

[Schoonderwoerd 96] Schoonderwoerd, R., Holland, O.E., Bruten, J.L., & Rothkrantz, L.J.M. (1996) *Ant-based load balancing in telecommunications networks*, HP Technical Report HPL-96-76.

***Tony White** has an M.A. (Cambridge, 81) in Theoretical Physics and an M.CS. (Carleton, 93). He is currently involved in collective problem solving research exploiting social insect metaphors in the Department of Systems Engineering at Carleton. Since 1986, he has been part of scientific staff at Nortel Technology and has been involved in issues relating to network management and design. His work has focused on the areas of Fault Diagnosis and optimization problems in Telecommunications. He was the principal architect of the Expert Advisor, a real time alarm correlation expert system. He is currently on secondment at the National Research Centre where his research activities are related to multi-agent systems for the management and control of heterogeneous networks. E-mail: twhite@ai.iit.nrc.ca, Telephone: (613) 998-7749*

Report on DX-96 The Seventh International Workshop on Principles of Diagnosis

Val Morin, Quebec October 13-16, 1996

Sue Abu-Hakima

DX est un atelier tenu annuellement dans le but d'encourager l'interaction et la coopération entre les chercheurs en intelligence artificielle avec des approches diverses face au diagnostic. Les ateliers précédents de cette série ont été tenus à Goslar (Allemagne) en 1995, à New York (ÉUA) en 1994, à Washington State (ÉUA) en 1992, à Milan (Italie) en 1991, à Stanford (ÉUA) en 1990 et à Paris (France) en 1989 (le 0ème atelier). Cette année, l'atelier se tenait à Val Morin dans les Laurentides, près de Montréal, Québec, Canada, et a consisté en quatre journées de présentations et de discussions.

DX is an annual workshop held to encourage interaction and cooperation among researchers in artificial intelligence with diverse approaches to diagnosis. Previous workshops in this series were held in Goslar (Germany) in 1995, New York (USA) in 1994, Washington State (USA) in 1992, Milan (Italy) in 1991, Stanford University (USA) in 1990, and Paris (France) in 1989 (the 0th one). This year, the workshop was held in Val Morin in the Laurentian Mountains near Montreal, Quebec, Canada and included four days of presentations and discussion.

Organization

The workshop was co-chaired by Sue Abu-Hakima (Canada), Gautam Biswas (USA), and Rene Bakker (Netherlands). The Program Committee included:

Mike Chantler (Heriot-Watt University, Scotland)
Luca Chittaro (Universita di Udine, Italy)
Marie-Odile Cordier (IRISA, France)
Kourosh Danai (University of Massachusetts, USA)
Adnan Darwiche (Rockwell, USA)
George Drastal (Siemens Corporation, USA)
Oskar Dressler (Occ'm Software GmbH, Germany)
Gerhard Friedrich (Siemens, Austria)
Rob Milne (Intelligent Applications, Scotland)
Amit Misra (Vanderbilt University, USA)
Wolfgang Nejdl (University of Hannover, Germany)
David Poole (University of British Columbia, Canada)
Lyle Ungar (University of Pennsylvania, USA)
Takashi Washio (Mitsubishi Research Institute, Japan)

Main Track Program

Fourty- six main track papers were submitted from thirteen countries including Australia, Austria, Canada, Finland, France, Germany, Italy, Japan, the Netherlands, Portugal, Spain, the UK, and the USA. Each paper had a minimum of three reviews and in borderline cases, a fourth review was undertaken. Papers that were strongly accepted were accepted for presentation and papers in earlier stages of work were accepted as posters. Fifteen papers were accepted for presentation and twelve accepted as posters.

As is usual with DX papers, some researchers focused on the theory of qualitative diagnosis and model-based reasoning while others emphasized the real-world implications and scalability of their approaches.

Highlights included Sqalli's paper on the use of constraint satisfaction techniques for testing LAN emulation over ATM networks. The idea is to receive a packet and treat it as an event and then map it to the constraint model. The only drawback to this approach was the absence of temporal knowledge which is quite significant in networks and their management.

McIlraith presented a very interesting paper on her work of integrating a system description with actions on the system for its diagnosis, testing, and repair. She described a four step process to integrate actions into the model-based representation. First, represent the system as a set of states for any part of the system that can change system behaviour. Second, distinguish state constraints into those that are based on ramification (e.g., causal knowledge), qualification (imposes constraint knowledge), or frame (situational) knowledge. Third, explicitly represent actions that will change the state description such as any repairs, replacements, or tests. Fourth, identify what is known as the initial state of the world.

Struss presented an interesting paper challenging the diagnosis of dynamic systems using state spaces versus simulation. This paper generated quite a bit of discussion amongst the researchers who believe that states cannot be truly monitored and as a result can never replace simulation.

Provan presented a paper on exploiting system structure to simplify computation in model-based diagnosis. He proposes the use of a causal structure which combines the actions on a physical system with the results from sensor data to simplify computation.

Williams and Nayak presented a paper on their work at NASA on the Cassini spacecraft which is to fly without humans to Mars. They described an interesting approach to the problem of configuration management should any of the subsystems fail in flight. The reason this approach is possible is that the subsystems have been isolated to make them causally independent.

For completeness, the fifteen presented papers, including those highlighted above, are as follows:

Sqalli, M.H., and Freuder, , E.C. "A constraint satisfaction model for testing emulated LANs in ATM networks," University of New Hampshire, USA.

Sabin, M. and Freuder, E.C. "Automated construction of constraint-based diagnosticians," University of New Hampshire, USA.

McIlraith, S. "SD + Actions: new representation problems for model-based diagnosis," University of Toronto, Canada.

Malik, A. and Struss, P. "Diagnosis of Dynamic Systems Does Not Necessarily Require Simulation," Technical University of Munich, Germany.

Sztipanovits, J. and Misra, A. "Diagnosis of Discrete Event Systems Using Ordered Binary Decision Diagrams," Vanderbilt University, USA.

Darwiche, A. and Provan, G. "Exploiting system structure in model-based diagnosis of discrete-event systems," Rockwell Science Center, California, USA

Williams, B. and Millar, B. "Automated decomposition of model-based learning problems," NASA Ames, California, USA

Williams, B. and Nayak, P.P. "A model-based approach to reactive self-configuring systems," NASA Ames, California, USA

Washio, T., Okusa, K., and Endou, A. "Intelligent virtual measurement in a process plant based on model-based diagnosis". Power Reactor & Nuclear Fuel Development Corp., Tsuruga, Japan

Chittaro, L. and Cortes, J.L. "Ship over troubled waters: functional reasoning with influences applied to the diagnosis of a marine technical system". University di Udine & University of Hamburg-Harburg, Italy & Germany

Cascio, F. and Sanseverino, M. "IDEA: Integrated diagnostic expert assistant - model-based diagnosis in the car repair centers," Fiat Research, Torino, Italy

ten Teije, A. and van Harmelen, F. "Examples of approximations in diagnosis based on approximate entailment," University of Amsterdam, the Netherlands

Chantler, M.J., Daus, S., Vikatos, T., and Coghill, G.M. "The use of quantitative dynamic models and dependency recording for diagnosis," Heriott-Watt University, Edinburgh, UK

D'Ambrosio, B. "Learning to diagnose using qualitative belief spaces," Oregon State University, USA

Brusoni, V., Console, L., Terenziani, P., and Dupre, D.T. "A spectrum of definitions for temporal model-based diagnosis," University of Torino, Italy

Poster Papers

Poster session paper presenters were given five minutes to provide a quick summary of their work to entice DX participants to visit their posters. In addition, the first two hour poster session was combined with a wine and cheese which made for a very relaxing and successful tour of the posters in the session.

A highlight of the poster session included Price et al.'s poster on Failure Modes Effects Analysis (FMEA). This work is ongoing with Jaguar of the UK. It has been validated by breaking real cars. The idea is to form trees of the cause and effects of the failures. Multiple failures can generate up to twenty thousand candidates. FMEA is thus organized into function-based diagnosis. Editors will be used to re-arrange and annotate the trees. An inference engine is used to execute the knowledge in the tree. This work is strikingly similar to what is known in the literature as fault-based diagnosis and is similar in concept to what was implemented in the JETA system by NRC.

Burton Lee presented an interesting poster on a distributed agent-based architecture for diagnosis. He envisions the web as an obvious network for linking together distributed nodes requiring diagnostic agents. He supports this idea by pointing out that companies are building diagnostic engines into their equipment (for example, Tektronix has a printer with on-board diagnosis). He also described Stanford's project of Internet-based on-line diagnosis for mobile cars.

For completeness, the twelve posters including those highlighted above, are as follows:

Mosterman, P.J. and Biswas, G. "An integrated architecture for model-based diagnosis of dynamic physical systems," Vanderbilt University, Tennessee, USA.

Stylianou, M. "Fault detection and diagnosis for commercial chillers," Canada Center for Mineral and Energy Technology, Varennes, Quebec, Canada
Topic: Sensor pattern recognition for diagnosis

Scarl, E. "Testing Safely," Boeing Co., Huntville, Alabama, USA
Topic: Test generation & model-based diagnosis

Price, C., Wilson, M., and Cain, C. "Generating fault trees from FMEA," University of Wales, Aberystwyth, UK & Ford Research, Michigan, USA
Topic: Fault mode and effects analysis & automated fault tree generation

Lee, B. and Ishii, K. "Agent-based remote diagnostics of product populations across the full product lifecycle: an industrial multi-agent system approach in an electronic commerce network," Stanford University, USA
Topic:Multi-agent system architecture for diagnosis

Ishida, Y. "Active Diagnosis by immunity-based agent approach," Nara Institute of Science & Technology, Nara, Japan
Topic: Multi-agent diagnosis, fault/failure-based

Bos, A. "Errors violating simplifying modeling assumptions," Control Systems Dept., TNO-TPD-TU, the Netherlands topic: model-based diagnosis

Liberatore, P. "Off-line computing of fault diagnosis in system graphs," University of Rome, Italy

Submitted for a student award topic: computational complexity, formalizing diagnosis

Loiez, E. and Taillibert, P. "Polynomial temporal band sequences for analog diagnosis," ISEN & Dassault Electronique, France
topic: model-based diagnosis Submitted for best student paper.

Kitamura, Y., Ueda, M., Ikeda, M., and Mizoguchi, R. "Diagnosis based on fault model events," Osaka University, Osaka, Japan

topic: model-based diagnosis

Bottcher, C., Dague, P., and Tailibert, P. "Hidden interactions in analog circuits," Fraunhofer Institute, Karlsuhe, Germany & France
topic: model-based diagnosis

Nejdl, W., Frohlich, P. "Minimal model semantics for diagnosis - techniques and first benchmarks," University of Hannover, Germany

DX Debate

DX-96 featured an impromptu debate on model-based diagnosis versus control engineering. Kourosh Danai of the University of Massachusetts pointed out that traditional engineering approaches can be quite useful to the MBD community. He pointed out that helicopter gear boxes can and are diagnosed using failure-driven diagnosis. He also pointed out that Kalman filters are used for state estimation in diagnosis. In addition, the notion of observability of state spaces (described by Peter Struss in his paper) exists in control theory. He also pointed out that simulation is used to generate the numbers in between measurable states. Greg Provan of Rockwell continued the debate by arguing that measurement and physical system decomposition (while not elegant MBD approaches) are very practical in the real world.

Brian Williams of NASA pointed out that maybe six years ago the MBD community did not talk to control engineers. But today, he believes that the two communities share common problems and goals and MBD researchers should pick up more of the control engineering knowledge and collaborate with researchers in the traditional engineering milieu. Bechir El Ayeb had a very different point of view. He argued that control theory is very far from model-based diagnosis. He said that the former is concerned with building an artifact and makes use of polynomial models for diagnosis. He added that the latter is concerned with using MBD to diagnose an existing artifact. There was general disagreement by the DX participants with this view that control engineering and MBD researchers are that far apart.

Invited Presentation

An invited presentation was given by Mike Halasz. Mike is the leader of the Integrated Reasoning Group at the Institute for Information Technology, National Research Council of Canada. His talk was entitled: "Lessons Learned in IDS and the Ongoing Collaboration with GE and Air Canada." His group has been collaborating with Air Canada and General Electric on the maintenance of the Airbus fleet. His group is developing an aircraft maintenance system that integrates diagnostic techniques with troubleshooting manuals. He specifically described the use of case-based reasoning to analyze real-time aircraft maintenance data so that as an aircraft lands the technicians are ready with any parts or repairs they need to perform. He emphasized the need for diagnostic systems to seamlessly be integrated into an organizational information system which combines traditional database and inventory systems. This real-world environment has not traditionally drawn AI researchers but should draw more in the future as researchers look for hard problems to tackle with AI.

Reference Problem Track

In keeping with the desires of the DX-community to establish benchmarks and reference problems for evaluating the current capabilities of diagnosis systems and defining a roadmap for the next generation of systems, DX-96 introduced a new track focused on a discussion of Diagnostic Reference Problems. The objective was to collect comprehensive and complete (including test data) problem descriptions that allow other researchers to develop alternate solutions and perform comparative evaluations.

To ensure that the reference problem would make a definitive contribution to the community and the field, the papers had to include the following:

- motivation for the problem
- description of the physical system
- application area and domain(s) covered
- description of primary problem dimensions
- unique characteristics of the problem (e.g., intermittent faults, changing topology, real-time solutions)
- measurements and available data, their source, and their quality
- current work on modeling, developing diagnosis algorithms, description and evaluation of solutions, and a possible classification label for the problem.

The long-term goal of the diagnosis community is to build a repository of reference problems (similar to the Machine Learning repository of data sets) that would be freely available to diagnosis researchers and help them in comparing approaches. The reference problems are also likely to provide the medium by which real-world practitioners interact with researchers in the field.

Gautam Biswas chaired the reference problem track. Five papers were submitted as reference problems: two papers from Canada,one from France, one from Germany, and one from the Netherlands. Only two papers qualified for presentation in the reference problem track, namely:

Dague, P. and Taillibert, P. "The monostable: a reference problem for analog diagnosis," University Paris-Nord & Dassault Electronique, France

Abu-Hakima, S. "Main Fuel System for an Aircraft Engine: a diagnosis reference problem. Institute for Information Technology, NRC, Ottawa, Canada.

Dague's paper focused on a well-bounded but difficult analogue circuit problem with many states. Abu-Hakima's paper focused on the use of a real-world aircraft engine component (a main fuel system) as a reference problem for electromechanical systems that include complex components and feedback loops.

Peter Stuss lead an excellent discussion on the nature of reference problems that the DX community is seeking. He emphasized the need for such problems not for benchmarking and not as solutions but as problems representing the real world. In essence, a problem addressing a real application that in turn represents a class of problems. He also emphasized the need for generating reproducible experiments. He also pointed out that problem categories vary based on the high number of components, the complex structure, whether a system is dynamic or not, whether feedback is an issue, whether the topology is a changing one, whether spatial relations exist in the model, whether the system is non-deterministic, and whether or not any measurements can be made. He concluded by pointing out the need for a variety of classes of reference problems that would include networks, electromechanical systems,and electronic circuits (digital or analogue).

Award for Best Student Paper

The Institute for Information Technology of the National Research Council of Canada sponsored an award for the Best Student Paper presented at DX'96. The selection for this award was made by the Program Committee after all student paper presentations were made. Only papers in which the first author was a full-time student were eligible for this award. The student had to also present the paper at DX-96 to be considered for the award.

Ten student papers were submitted for the award. Of the ten, four papers were accepted as poster papers and three accepted as full papers thus qualifying for the student award. Of the three papers, only two papers were presented and hence two awards were given. The highest ranked student paper in terms of quality and presentation was the University of Toronto paper by Sheila McIlraith entitled: "SD + Actions: new representation problems for model-based diagnosis." The second paper award went to the student from the University of New Hampshire, Mohammed Sqalli for his paper entitled: " A constraint satisfaction model for testing emulated LANs in ATM networks (included in this issue of the magazine).

Conclusion

Fourty-one people participated in this highly successful four day workshop. The beautiful fall colors in the Laurentians and the Far Hills Inn kept attendees relaxed and focused on Diagnosis. I would like to take this opportunity to thank all those who participated and hope for us all to continue exchanging ideas on this exciting field.

Footnote: DX-96 Proceedings were published by NRC as a technical report and are available by request from suhayya@ai.iit.nrc.ca.

Sue Abu-Hakima is the Group Leader of the Seamless Personal Information Networking (SPIN) Group at the Institute of Information Technology at NRC. SPIN focuses on the research and development of advanced software including agent and diagnostic technologies to real-world problems in telecommunications. She has been a researcher at NRC since 1987. She had previously worked atBNR starting in 1982. She has Doctorate and Masters degrees in AI fromCarleton University's Department of Systems and Computer Engineering. She also has a B. Eng from McGill University's Department of Electrical Engineering with specialties in Communications and Computers.

Book Reviews

The robot's dilemma revisited: The frame problem in artificial intelligence *Kenneth M. Ford and Zenon W. Pylyshyn (editors)* (Rutgers University and University of West Florida) Norwood, NJ: Ablex Publishing Corporation, 1996, xviii+141pp; hardbound, ISBN 1-56750-142-7; paperbound, ISBN 1-56750-143-5

Reviewed by
Javier Pinto
Catholic University of Chile

The book is a collection of papers that evolved from talks presented at The First International Workshop on Human and Machine Cognition, held in Pensacola Beach, Florida in May 1989. The editors point out that the published papers are much more recent, and were completed after the workshop "(in some cases *far* after)." The editors of the book note that the workshop articles came in two flavours. There were a number of articles with a "computer science (logic) flavour," which were published in a separate volume (Ford and Hayes 1991). The other articles, which appear in the present book, are of a more "philosophical" flavor. Therefore, the book is highly biased in the style of articles presented. There is a strange exception, that of Leora Morgerstern's article. I find it strange because the article has a strong logic flavor (I will have more to say about the article later).

Almost thirty years have passed since McCarthy and Hayes first described the frame problem. Ever since, the frame problem has motivated a great deal of research in artificial intelligence and in cognitive science. It has also created a great deal of confusion and exasperation. The problem arises when one attempts to describe the effect that actions or events have on the world by using mathematical logic. Thus, it is rather perplexing that it was decided to separate articles with a *logic* flavor from the others. After this separation, it is legitimate to wonder whether the frame problem for computer scientists is different from the frame problem for philosophers. However, after reading the material, one gets the feeling that one is dealing more or less with the same problem. The separation is also very disappointing, since one would expect that a workshop on the frame problem would be a great opportunity to provide bridges between differing viewpoints on the problem. A case in point is those articles that refer to one of Fodor's examples in which the concept of a *fridgeon* is introduced. The example goes as follows: All particles are fridgeons if and only if Fodor's refrigerator is open. Fodor's problem is meant to illustrate that to represent the concept of a *fridgeon* one needs to write a tremendous number of axioms. However, I would venture that any logicist would quickly solve this problem using some first-order temporal logic. For instance, Shanahan, in his book *Solving the frame problem: A mathematical investigation of the common sense law of inertia* (to appear), shows how trivial this problem really is. However, Fodor's example shows up in several articles in the collection. Maybe this problem would go away if there were some cross-communication between the computer scientists and the philosophers (using the categories introduced by the book editors). Another source of disappointment is that, with the exception of Morgernstern's article, all references to the logicist research on the frame problem were published before the 1989 workshop.

The book contains eight articles, plus an introduction and an epilogue. Some articles propose general ideas on how one could deal with the frame problem without using a logical framework. For instance, Daniel Dennett proposes to use stories in order to predict the future. Thus, in order to predict what is going to happen as a result of some action, one would re-enact an appropriate story. Lars-Erik Janlert, on the other hand, proposes the use of *pictures* to deal with the frame problem. My understanding is that a picture is some analogical model of reality, and reasoning about effects of actions can be done by *simulating* the execution of the action in the analogical model. A hybrid system that employs a *verbal* representation and pictures would be the basis for a system that solves the frame problem. Another proposal is that of Kyburg, who believes that reasoning about change in dynamic worlds should involve probabilities, and that one should use utilities in order to decide how to proceed. Another perspective is provided by Glymour, who points out that the frame problem poses questions about the prediction of the effects of actions in a causal system. Furthermore, he regrets that these questions have been largely ignored in AI, with some exceptions. The exceptions deal mostly with inferring causal structure from data.

The other articles do not propose or discuss specific approaches to deal with the frame problem. For instance, Dietrich and Fields claim that cognitive scientists should not try to explain human intelligence as approximating ideal rationality. Furthermore, he suggests that a proper viewpoint is biology. This article also presents a very interesting discussion on the role that the frame problem plays in cognitive science. In general, it seems to me, this article is not really about the frame problem; rather it deals with the direction that cognitive science should take.

Lormand suggests that one should be careful in studying the frame problem, since there are other related problems, namely the *relevance* problem and the *holistic* problem, "... which have nothing important to do with the frame problem as it is understood in AI." Furthermore, he discusses how some researchers wrongly attack strategies to deal with the frame problem (the *sleeping dog* strategy), pointing towards difficulties associated with the other two problems. Thus, the arguments against these strategies are improper. Finally, Lormand discusses a way to structure memory in order to deal with *kooky* predicates, like Fodor's *fridgeon* predicate. Lormand's suggestion is to keep the definition of *fridgeon* in memory, and use the definition to determine if something is a fridgeon. Interestingly, this proposition is compatible with what some logicists have proposed with regards to the ramification problem. This is one instance where there is a clear need for communication between *computer scientists* and *philosophers*.

Loui wrote a piece in which he looked at the original collection of articles on the frame problem under the title *The Robot's Dilemma* (Pylyshyn 1987). Loui discusses whether or not the authors' ideas survive after the introduction of the *Yale Shooting Problem*. Loui provides a very interesting discussion. However, the most important point he raises, I believe, is that one wants "... to produce easily specified, easily manipulated, flexible and accurate, comprehensible representations that can be implemented on a computer." Furthermore, Loui wonders whether some of the musings in the original set of articles were going to be of any help to AI. The same question is, in my opinion, valid with respect to the volume under discussion.

The eighth article is the odd one in the group, since it contains explicit logic. This article, by Leora Morgenstern, a confessed logicist, is an overview of the state of the art regarding the logicist work on the frame problem (ca. 1993). Morgenstern presents the point of view that logicist research has been somehow flawed and that one needs to suggest a new course of research in order to make progress. This article is very nicely written. Unfortunately, it ignores research on the frame problem that has been published after 1993. In spite of this, Morgenstern raises valid points. In particular, she suggests that we need a new trend in the methodology used in logicist AI research. This new trend should be the use of existing theories, to build upon existing research, and to modify and integrate. As Morgenstern recognizes, some researchers (e.g., Lifschitz, Sandewall) have already contributed towards this trend. I believe that a great deal of effort is being invested in this direction already.

In conclusion, is this book worth reading? From my perspective, admittedly logicist, the main drawback of the book is that it is a little outdated. However, there are many ideas that are quite relevant. While reading the articles in the book, I was at times confused, exasperated, and amused. All these feelings are consonant with the topic of the infamous *frame problem*. I would not advise any neophyte to read this book as a means to get acquainted with the problem. This book should be interesting for somebody already familiar with the frame problem. The book contains several amusing *bed time* stories that would help any insomniac logicist to get to sleep.

References

Ford, Kenneth M. and Hayes, Patrick J. (editors) (1991). *Reasoning agents in a dynamic world: The frame problem*. Greenwich, Conn.: JAI Press.

Pylyshyn, Zenon W. (editor) (1987). *The Robot's dilemma: The frame problem in artificial intelligence*. Norwood, N.J. : Ablex.

Books Received

Books marked with a + in the list below are scheduled for review in a future issue. Reviewers are still sought for books marked with a *. Readers who wish to review books for *Canadian Artificial Intelligence* should write, outlining their qualifications, to the book review editor, Graeme Hirst, Department of Computer Science, University of Toronto, Toronto, Canada M5S 3G4, or send electronic mail to gh@cs.toronto.edu or gh@cs.utoronto.ca. Obviously, we cannot promise the availability of books in anyone's exact area of interest.

Authors and publishers who wish their books to be considered for review in *Canadian Artificial Intelligence* should send a copy to the book review editor at the address above. All books received will be listed, but not all can be reviewed.

A course in generalized phrase structure grammar Paul Bennett (UMIST) London: UCL Press (Studies in computational linguistics, edited by Harold Somers), 1995, xi+227 pp; paperbound, ISBN 1-85728-217-5.

Corpus processing for lexical acquisition Branimir Boguraev and James Pustejovsky (editors) (Apple Computer and Brandeis University) Cambridge, MA: The MIT Press (Language, speech, and communication series), 1996, xvii+245 pp; hardbound, ISBN 0-262-02392-X.

Recent Advances in Parsing Technology Harry Bunt and Masaru Tomita (editors) (Tilburg University and Keio University) Dordrecht: Kluwer Academic Publishers (Text, speech and language technology series, edited by Nancy Ide and Jean Véronis, number 1), 1996, xv+415 pp; hardbound, ISBN 0-7923-4152-X, US$128.00

Linguistic Instruments in Requirements Engineering J.F.M. Burg (Vrije Universiteit, Amsterdam) Amsterdam: IOS Press, 1997, xii+296 pp; hardbound, ISBN 90-5199-316-1.

The cerebral code: Thinking a thought in the mosaics of mind William H. Calvin (University of Washington) Cambridge, MA: The MIT Press, 1996, v+256 pp; hardbound, ISBN 0-262-03241-4, US$22.50

Linguistics and computation Jennifer Cole, Georgia M. I. Illinois at Urbana-Champaign) Stanford, CA: Center for the

Study of Language and Information (CSLI lecture notes 52), distributed by Cambridge University Press, 1995, xi+296 pp; hardbound, ISBN 1-881526-82-8, US$49.95; paperbound, ISBN 1-881526-81-X, US$22.95

Linguistic concepts and methods in CSCW John H. Connolly and Lyn Pemberton (editors) (Loughborough University and University of Brighton) London: Springer (Computer supported cooperative work series, edited by Dan Diaper and Colston Sanger), 1996, xvii+230 pp; paperbound, ISBN 3-540-19984-5

Prolog programming in depth Michael A. Covington, Donald Nute, and André Vellino (University of Georgia, University of Georgia, and Nortel Technologies) Upper Saddle River, NJ: Prentice Hall, 1997, x+516 pp; paperbound, ISBN 0-13-138645-X, no price listed

Partiality, Modality, and Nonmonotonicity Patrick Doherty (editor) (Linköping University) Stanford, CA: CSLI Publications and The European Association for Logic, Language and Information (Studies in Logic, Language and Information), 1996, x+300 pp; distributed by Cambridge University Press; hardbound, ISBN 1-57586-031-7, US$64.95; paperbound, ISBN 1-57586-030-9, US$22.95

Intelligent language tutors: Theory shaping technology V. Melissa Holland, Jonathan D. Kaplan, Michelle R. Sams (editors) (U.S. Army Research Institute) Mahwah, NJ: Lawrence Erlbaum Associates, 1995, xviii+384 pp; hardbound, ISBN 0-8058-1502-3, US$79.95; paperbound, 0-8058-1503-1, US$39.95

Computers as assistants: A new generation of support systems Peter Hoschka (editor) (GMD, Sankt Augustin) Mahwah, NJ: Lawrence Erlbaum Associates (Computers, cognition, and work series, edited by Gary M. Olson, Judith S. Olson, and Bill Curtis), 1996, vii+313 pp; paperbound, ISBN 0-8058-2188-0, US$29.95; hardbound, ISBN 0-8058-2187-2, US$59.95

Analogical natural language processing Daniel Jones (UMIST) London: UCL Press (Studies in computational linguistics, edited by Harold Somers), 1996, xii+155 pp; hardbound, ISBN 1-85728-218-3.

Quantifiers, deduction, and context Makoto Kanazawa, Christopher J. Piñon, and Hentiöette de Swart (editors) (Chiba University; Max-Planck-Gesellschaft, Berlin; and Stanford University) Stanford, CA: Center for the Study of Language and Information (CSLI lecture notes 57), distributed by Cambridge University Press, 1996, x+197 pp; hardbound, ISBN 1-57586-005-8, US$49.95; paperbound, ISBN 1-57586-004-X, US$22.95

The Balancing Act: Combining Symbolic and Statistical Approaches to Language Judith L. Klavans and Philip Resnik (editors) (Columbia University and University of Maryland) Cambridge, MA: The MIT Press (Language, speech, and communication series), 1996, xiv+186 pp; hardbound, ISBN 0-262-11218-3, US$35.00; paperbound, ISBN 0-262-61122-8, US$17.50

Semantics and pragmatics of natural language: Logical and computational aspects [Proceedings of the Donostia—Toulouse 1993 Workshop] Kepa Korta and Jesús M. Larrazabal (editors) (University of the Basque Country) San Sebastián: Institute for Logic, Cognition, and Language, University of the Basque Country, 1995, 248 pp; paperbound, ISBN 84-920104-3-6, 2100 ptas (order from ilredito@sk.ehu.es)

Syntactica (NeXTStep edition) Richard K. Larson, David S. Warren, Juliana Freire de Lima e Silva, and Konstantinos Sagonas (State University of New York, Stony Brook) Cambridge, MA: The MIT Press, 1996, x+130 pp and 3.5-inch diskette for PCs and NeXTStations running NeXTStep 3.2 or higher; hardbound, ISBN 0-262-62106-1, US$25.00

Using speech recognition Judith A. Markowitz Upper Saddle River, NJ: Prentice Hall PTR, 1996, xxii+292 pp; paperbound, ISBN 0-13-186321-5, US$34.00

Integration of natural language and vision processing, Volume II: Intelligent multimedia Paul Mc Kevitt (editor) (University of Sheffield) [Reprinted from *Artificial Intelligence Review*, 9(2-3), 1995.] Dordrecht: Kluwer Academic Publishers, 1995, v+170 pp; hardbound, ISBN 0-7923-3758-1, US$66.00.

Parsing with Principles and Classes of Information Paola Merlo (University of Geneva) Dordrecht: Kluwer Academic Publishers (Studies in linguistics and philosophy, edited by Gennaro Chierchia, Pauline Jacobson, and Francis J. Pelletier, number 63), 1996, x+245 pp; hardbound, ISBN 0-7923-4103-1, US$90.00.

Design rationale: Concepts, techniques, and use Thomas P. Moran and John M. Carroll (editors) (Xerox Palo Alto Research Center and Virginia Polytechnic Institute and State University) Mahwah, NJ: Lawrence Erlbaum Associates (Computers, cognition, and work series, edited by Gary M. Olson, Judith S. Olson, and Bill Curtis), 1996, vi+488 pp; hardbound, ISBN 0-8058-1566-X, US$99.95; paperbound, 0-8058-1567-8, US$49.95

(continued on page 28)

5th Pacific Rim International Conference on Artificial Intelligence (PRICAI) 22 - 27th November, 1998 Singapore

General Information

The Pacific Rim International Conferences on Artificial Intelligence (PRICAI) are biennial international events which concentrate on AI technologies and their applications in areas of social and economic importance for countries in the Pacific Rim. The objective of PRICAI is to promote research and development of artificial intelligence in Pacific Rim countries by:

- providing a forum for the introduction and discussion of new research results, concepts, and technologies;
- providing practicing engineers exposure to and an evaluation of evolving research, tools, and practices;
- providing the research community exposure to the problems of practical applications of artificial intelligence; and
- encouraging the exchange of artificial intelligence technologies and experience within Pacific Rim countries.

This conference attempts to meet the needs of a large and diverse constituency, which includes practitioners, researchers, educators, and users. The focus of the conference is on areas of artificial intelligence that of particular importance in Pacific Rim countries.

The PRICAI conferences are managed by a Steering Committee (currently chaired by Michael Georgeff, Australia, Institute of Artificial Intelligence), and a Scientific Committee (currently chaired by Randy Goebel, University of Alberta). Each individual conference is managed by an Organizing Committee, and a Program Committee. The current structure and operation of PRICAI, and the steering and scientific committees are specified by the PRICAI Conference bylaws.

The first PRICAI conference was held in Nagoya, Japan in 1990. Subsequent conferences have been in Seoul, Korea (1992), Beijing, China (1994), and Cairns, Australia (1996). The conference has grown, both in participation and scope, and now includes participation from all major Pacific Rim nations. The conference also attracts participation from all over the world, including Europe, Africa, India, and South America.

Singapore, 22 - 27 November 1998
Organised By:
Japan-Singapore AI Centre
Information Technology Institute
Institute of Systems Science
National University of Singapore

In Cooperation With:

- National Computer Board
- National Science & Technology Board
- Singapore Computer Society
- Canadian Society for Computational Studies of Intelligence
- Japanese Society for Artificial Intelligence
- Taiwanese Association for AI
- Instrumentation and Control Society of Malaysia
- Intelligent Automation Society of the Chinese Association of Automation

The Pacific Rim International Conferences on Artificial Intelligence (PRICAI) are biennial international events which concentrate on AI technologies and their applications in areas of social and economic importance for countries in the Pacific Rim. The objective of PRICAI is to promote research and development of AI in Pacific Rim countries by:

- providing a forum for the introduction and discussion of new research results, concepts, & technologies;
- providing practising engineers exposure to and an evaluation of evolving research, tools, and practices;
- providing the research community exposure to the problems of practical applications of AI; and
- encouraging the exchange of AI technologies and experience within Pacific Rim countries.

This conference attempts to meet the needs of a large and diverse constituency, which includes practitioners, researchers, educators, and users. The focus of the conference is on areas of AI that of particular importance in Pacific Rim countries.

Submission

Papers describing both theoretical and applied research in AI can now be submitted for PRICAI '98. Topics of interest include, but are not limited to:

- AI research and development having, or are anticipated to have impact in the Pacific Rim (e.g., intelligent scheduling, multilingual text and language processing, financial modelling, intelligent manufacturing)
- AI research of a theoretical nature, but with some vision of its potential impact on applications (e.g., planning for intelligent robotics, belief revision for scheduling, induction and learning for data abstraction and data mining)
- AI research and development in the software development process, including information and software repositories, software analysis and abstraction, enterprise modelling)

Topics which are having, or are anticipated to have impact within the Pacific Rim, are of special interest. ***Papers on commercial or industrial applications of AI are especially encouraged.***

Please submit 5 hard copies of your paper to:

PRICAI '98 Secretariat
Japan-Singapore AI Centre, Information Technology Institute
11 Science Park Road, Singapore Science Park II, Singapore 117685
tel: (65) 770 5080
fax: (65) 779 1827
email: pricai98@iti.gov.sg

Multiple submission policy for papers

Papers that are being submitted to other AI conferences, whether verbatim or in essence, must reflect this fact on the title page. If a paper is accepted at another conference (with the exception of specialized workshops), it must be withdrawn from PRICAI '98. Papers that do not meet these requirements are subject to rejection without review.

Call for Tutorial Proposals

Proposals for tutorials in specific fields of AI can now be submitted for PRICAI '98. Details can be obtained from the PRICAI '98 web page.

Call for Workshop Proposals

Proposals for workshops in specific fields of AI can now be submitted for PRICAI '98. Details can be obtained from the PRICAI '98 web page.

Important Dates

10 April 1998 – Submission of papers, tutorials & workshop proposals
19 June 1998 – Notification of acceptance
31 July 1998 – Final camera ready paper

Review

All submissions will be reviewed on the basis of relevance, originality, significance, soundness and clarity. At least two referees will review each submission independently.

Publication

Papers accepted for presentation at PRICAI '98 will appear in a proceedings produced by an international publisher.

Contact Information

General information
PRICAI '98 Secretariat (pricai98@iti.gov.sg) tel: (65) 770 5080, fax: (65) 779 1827

Sponsorship & industry participation:
Lynica Foo (lynica@iti.gov.sg)
tel: (65) 770 5948, fax: (65) 779 1827.

Books Received

(con't. from page 25)

Doing worlds with words: Formal semantics without formal metaphysics Jaroslav Peregrin (Charles University) Dordrecht: Kluwer Academic Publishers (Synthese library, edited by Jaakko Hintikka, volume 253), 1995, xvi+236 pp; ISBN 0-7923-3742-5, US$97.00.

The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places Byron Reeves and Clifford Nass (Stanford University) Stanford, CA: CSLI Publications and Cambridge University Press, 1996, xiv+305 pp; hardbound, ISBN 0-57586-052-X, US$27.95

The human semantic potential: Spatial language and constrained connectionism Terry Regier (University of Chicago) Cambridge, MA: The MIT Press (Neural network modeling and connectionism series, edited by Jeffrey L. Elman), 1996, xv+220 pp; hardbound, ISBN 0-262-18173-8, US$37.50

Applications of Natural Language to Information Systems [Proceedings of the Second International Workshop, Amsterdam, 26-28 June 1996] R.P. van de Riet, J.F.M. Burg, and A.J. van der Vos (editors) (Vrije Universiteit, Amsterdam) Amsterdam: IOS Press, 1996, x+277 pp; hardbound, ISBN 90-5199-273-4, US$94.00.

Phrase structure and the lexicon Johan Rooryck and Laurie Zaring (editors) (University of Leiden and Macalester College) Dordrecht: Kluwer Academic Publishers (Studies in natural language and linguistic theory, edited by Liliane Haegeman, Joan Maling, and James McCloskey, volume 33), 1996, vii+298 pp; hardbound, ISBN 0-7923-3745-X, US$99.00.

The new writing environment: Writers at work in a world of technology Mike Sharples and Thea van der Geest (editors) (University of Sussex and University of Twente) London: Springer, 1996, x+274 pp; paperbound, ISBN 3-540-76011-3.

Internet Dreams: Archetypes, Myths, and Metaphors Mark Stefik (editor) (Xerox Palo Alto Research Center) Cambridge, MA: The MIT Press, 1996, xxiv+412 pp; hardbound, ISBN 0-262-19373-6, US$30.00

HAL's Legacy: *2001*'s Computer as Dream and Reality David G. Stork (editor) (Ricoh California Research Center) Cambridge, MA: The MIT Press, 1997, xxi+384 pp; hardbound, ISBN 0-262-19378-7, US$22.50

Communication Patterns and Textual Forms Graziella Tonfoni (University of Bologna) Exeter: Intellect Books, 1996, 123 pp; paperbound, ISBN 1-871516-65-X.

Connectionist, statistical, and symbolic approaches to learning for natural language processing Stefan Wermter, Ellen Riloff, and Gabriele Scheler (editors) (University of Hamburg; University of Utah; and Technical University of Munich) Berlin: Springer (Lecture notes in artificial intelligence 1040), 1996, ix+468 pp; paperbound, ISBN 3-540-60925-3, no price listed

Linguistic and computational techniques in machine translation system design (2nd edition) Peter Whitelock and the late Kieran Kilby (Sharp Laboratories of Europe and UMIST) London: UCL Press (Studies in computational linguistics, edited by Harold Somers), 1995, xii+208 pp; hardbound, ISBN 1-85728-216-7.

Electric words: Dictionaries, computers, and meanings Yorick A. Wilks, Brian M. Slator, and Louise M. Guthrie (University of Sheffield, Northwestern University, and Lockheed Martin) Cambridge, MA: The MIT Press (ACL-MIT Press series in natural language processing, edited by Aravind K. Joshi, Karen Sparck Jones, and Mark Y. Liberman), 1996, ix+289 pp; hardbound, ISBN 0-262-23182-4, US$32.50