VOLUME 2

PRICAI '94

THE 3rd PACIFIC RIM INTERNATIONAL CONFERENCE
ON ARTIFICIAL INTELLIGENCE

BEIJING CHINA ◇ AUGUST 15-18, 1994

SPONSORED BY CHINA COMPUTER FEDERATION
CHINESE ASSOCIATION OF AUTOMATION

PRICAI-94

Proceedings of the 3rd
Pacific Rim International Conference
on Artificial Intelligence

August 15-18, 1994
Beijing, China

Volume 2

Sponsored by
China Computer Federation
Chinese Association of Automation

Co-sponsored by
IFIP TC 12
IEEE Computer Society
American Association for Artificial Intelligence(AAAI)
Japanese Society for Artificial Intelligence(JSAI)
Korea Information Science Society
Hong Kong Computer Society
Chinese Association for Artificial Intelligence(CAAI)
Singapore Computer Society

Supported by
State Communication of Science and Technology, China
National Natural Science Foundation of China
Chinese Academy of Sciences

INTERNATIONAL ACADEMIC PUBLISHERS

(京)新登字 141 号

图书在版编目(CIP)数据

人工智能：太平洋国际人工智能会议论文集：英文／史忠植编. - 北京：万国学术出版社，1994.8
ISBN 7-80003-319-8

I.人… II.史… III.人工智能-国际会议-文集-英文
IV.TP18-53

中国版本图书馆 CIP 数据核字 (94) 第 06989 号

Published and Distributed by
International Academic Publishers
137 Chaonei Dajie, Beijing 100010
The People's Republic of China

The book has been photographically reproduced from the best available manuscripts. The papers were not refereed but were reviewed for their technical contents. Editing was restricted to matters of format, general organization and retyping.
The editors assume no responsibility for the accuracy, completeness or usefulness of the information disclosed in this volume. Unauthorized use might infringe on privately owned patents of publication right. Please contact the individual authors for permission to reprint or otherwise use information from their papers.

First edition 1994

Shi Zhongzhi (史忠植)
Proceedings of the 3rd Pacific Rim International Conference on Artificial Intelligence Vol. 2
太平洋国际人工智能会议论文集　第二卷

ISBN 7-80003-319-8 / TP・11

Printed by the Printing House of China Building Industry Press

PREFACE

The Pacific Rim countries are undergoing rapid social and economic change. Advanced computing technologies, particularly artificial intelligence, offer the potential to greatly benefit these societies. This conference will focus on AI technologies and applications in areas of social and economic importance for countries in the Pacific Rim, particularly those of value to the host country and its neighbours.

It has been an honor and a pleasure to organize the program for the Third Pacific Rim International Conference on Artificial Intelligence to be held at Friendship Hotel in Beijing, China, during August 15-18, 1994. We wish to express our gratitude to the researchers in the field who have responded with enthusiam by submitting such high quality papers. We received 308 manuscripts from 17 countries. The topic area distribution for the papers appearing in the Proceedings is below:

Architectures for AI	4
Automated reasoning	17
Cognitive modelling	3
Distributed AI	5
Knowledge Base Technology	9
Knowledge representation	19
Machine Learning	23
Natural language processing	22
Neural Networks	13
Pattern Recognition	11
Computer Vision	20
AI Applications	19

We regret that many of high quality papers had to be turned down due to the physical constraints. Each submitted paper was reviewed by two referees designated by the program committee. The Program Chairs made the final selection accepting 165 papers for this presentation.

We hope that you will find this conference program—the invited speech, the panels, the tutorials, the workshop, and the individual paper presentations— exciting, stimulating and illuminating. This program could not have happend if not for dedication of many of you, the contributors, the Conference Committee, the Program Committee, the reviewers. We are extremely grateful to colleagues who made the PRICAI possible and successful.

Program Chairs
Michael Georgeff
Zhongzhi Shi
Benjamin W. Wah

CONFERENCE COMMITTEE

CONFERENCE CHAIR

Ruwei Dai (China)

PROGRAM COMMITTEE

Chairs:

Michael Georgeff (Australia)
Zhongzhi Shi (China)
Benjamin W. Wah (USA)

Members:

Vasant Dhar (USA)
Timothy Finin (USA)
Michael Georgeff (Australia)
Randy Goebel (Canada)
Zhijun He (China)
Cheng-Seen Ho (Taiwan, China)
Se June Hong(USA)
David Israel(USA)
Jin Hyung Kim(Korea)
Young-Bin Kwon (Korea)
Kwong Sak Leung(Hong Kong)
Young-Hwan Lim (Korea)
Xuhua Liu(China)
Ruqian Lu(China)
Fumio Mizoguchi(Japan)
Riichiro Mizoguchi(Japan)
Juzar Motiwalla(Singapore)
Jeff Rosenschein (Israel)
R. Sadananda(Thailand)
Peter Selfridge (USA)
B. Shekar(India)
Chunyi Shi(China)
Zhongzhi Shi(China)
Ching Y. Suen(Canada)
Mario Tokoro (Japan)
Saburo Tsuji (Japan)
Benjamin W. Wah (USA)
Vilas Wuwongse (Thailand)
Wai K. Yeap (New Zealand)
Bo Zhang (China)
Chengqi Zhang (Australia)
Jan M. Zytkow (USA)

ORGANIZATION CHAIR

Yaotong Li (China)

EXHIBITION COMMITTEE CHAIR

Shukai Chen (China)

CONFERENCE SECRETARIAT

Guanghua Li (China)

CONTENTS

VOLUME 1

5. MACHINE LEARNING

6. NEURAL NETWORKS

VOLUME 2

10. PATTERN RECOGNITION

11. COMPUTER VISION

12. AI APPLICATIONS

7. ARCHITECTURE FOR AI

DISTRIBUTED SYSTEMS: DATA OR CONTROL ?

Bernard Burg, François Arlabosse
FRAMENTEC-COGNITECH
Tour Fiat cedex 16
92084 Paris la Défense, France
archon@framentec.fr

ABSTRACT

The ARCHON **AR***chitecture for* **C***ooperative* **H***eterogeneous* **ON***-line systems* project [15] aims to provide a distributed control system over existing industrial applications such as telecommunication or electric networks. These applications face the problems of :
- distributed control since the control is spread throughout networks.
- distributed data because the data handled in those applications are often complex, having several levels and embedded structures present in different parts of the network.

This paper explains how ARCHON handles those problems by using independent mechanisms for the control and the data, both of which are driven by a cooperation mechanism. This leads to a review of the existing solutions and the setting of goals for a new control architecture. The ARCHON architecture is then described and we show how it answers the original problem. The conclusions show the added value brought to the Open System Foundation.

INTRODUCTION

In the previous papers, ARCHON has been described as a distributed control architecture where each agent cooperates to achieve of a global goal without local explicit knowledge of it. As such, the System Restoration Planning test-bed has always been described by its sole control aspects whereas the data management has never been described in the project framework.

Yet, quite often in the bibliography, distributed systems are analyzed through two dimensions - control and the data management. We use these same dimensions to introduce examples with the type of problems ARCHON can handle. This paper describes the implemented mechanisms and focuses on the distributed control as well as on the Federated Database System AIM, which provides new database handling facilities such as personalized viewpoints on data for each agent. An example of a cooperation scenario is presented so as to demonstrate the design of a control platform.

1 Distributed Control

Plant-wide solutions for the problem of control require a combination of programmable controller and distributed control systems capabilities[11,17], which involve a hybrid of discrete and wet-flow processing functions. Due to the different constraints at the levels of the CIM pyramid, those functions are carried out by different machines linked in distributed control systems (DCSs). At the lowest level, sensors and actuators are connected to PLCs: The PLCs are connected through industrial networks which in turn, may be connected to computer networks.

The decentralized control brings both: the *scalability* and *openness* of systems to cope with the growing complexity of the application. In such a context, the application can be built block by block, which allows a progressive investment and a step-by-step debug and tuning of the application.

In addition, the decentralized approach favors robust and fault tolerant controllers since the control does not rely on a single site but is shared amongst many network elements. One can conceive systems insuring their functions for the best, even in degenerated cases when some parts of the controlled installation are faulty.

2 Distributed Data

Open Information Systems [10] (OIS) are large-scale information systems that are always subject to asynchrony in input and output. A good example of such systems are enterprise-wide information systems embedding the information of the manufacturing, Research and Development, Sales and Marketing departments. In many cases, those OIS systems are geographically distributed, over long periods of time and have dependencies on overlapping sub-projects.

Some problems, as the access to a shared financial account via multiple sites using electronic funds transfer, show the limitations of the *Deductive Logic*, since no amount of knowledge of the physical circumstances in which the withdraw requests are made determines the outcome. In such cases some control mechanisms have to be introduced in addition to the distributed data-base.

The problem of the shared bank account access is easily solved by a *serializer*, which allows only one action at the time. Then the data-base problem is to access the data structures through schemas, and to retrieve, write and modify them perhaps adding new properties.

The separation between distributed control and distributed data allows to master the complexity of both functions but requires the cooperation between control and data so to keep the system properly updated.

3 Distributed Control and Distributed Data

Some systems, like electrical or telecommunication networks are by definition geographically distributed. Their control is as such implemented in a distributed fashion to provide modularity and fault tolerance.

When dealing with applications like Demand Management, Alarm Analysis or System Restoration Planning, the problem is twofold: control as well as data management. During an alarm analysis, the system supervisor needs both kinds of information:
- *alarm messages* describe the temporal events reported in the network. They give an image of the history and the current state of the network. They are, in general short and simple (no embedded structures), but their temporal aspect and their order of arrival are relevant. The response time of their treatment is extremely important.
- *network description:* describes the layout of the network, its components and properties. This gives the static view of the network. They are often long (sometimes several Mega-bytes) and complex structures, describing graphs, subgraphs, properties... As this information is static, there are low temporal constraints.

The nature of the information carried by both types of messages is different. Thus each of them needs an adapted support and has to be processed in accordance with their very own constraints. The *Alarm messages* correspond to the control architecture, the messages are short but need to be processed in a very efficient manner so as to allow the system to perform control over applications. On the other hand, the *network description* is used to transmit complex and/or bulky data in a structured manner. It is important to provide tools and facilities to transmit and retrieve this data in a network, so that one can access several data-bases in a transparent manner, even if different data-representations are used locally.

4 Example of Distributed Control and Distributed Data

An application scenario of the ARCHON test-bed was to perform System Restoration Planning[4]. The following is a simplified scenario. Each of the agents is in charge of solving a part of the problem and these agents are embedding existing *Intelligent Systems (ISs)*:

The *DIAGNOSIS agent* monitors the *Alarm messages* of the network. As soon as it detects an *Alarm message* indicating an error in a part of the network, the *DIAGNOSIS agent* has to determine the possible black-out area affected by the alarm. To do so, this agent requests from the *INFORMATION agent* the *Network description*, giving the topology of the network and its geographic mapping. Meanwhile the new *Alarm messages* have been stored. The *DIAGNOSIS agent* compares the *Alarm messages* and the *Network description* to determine the blackout area, by checking the position of the breakers (given by another agent). This calculated black-out area is then compared with customers' complaints gathered through the *PHONE agent* giving the time and area code of their phone-calls. On the basis of this information, the *DIAGNOSIS agent* determines the possible causes of the trouble and gives this information to the *Restoration Planner Agent.*

The control aspects of this application are shown by the *Alarm messages.* Their order of arrival is important to diagnose of the network to spot the causal effects. The difficulty arises because errors propagate in networks and examples of complete electricity distribution networks collapsing still remain present in our memories. Fast actions have to be undertaken.

The data management aspects are related to the *Network description.* They are encoded in the form of graphs with several layers of graphs embedded, according to geographic location, the nature of the electricity (High or Low voltage), the technology used... The complete network description may take several Mega-bytes, thus it is important to be able to select the parts of the network relevant to the work performed by the *DIAGNOSIS agent.* An object-oriented federated data-base provides tools to retrieve the relevant object or structure and to transmit it from one agent to another.

System restoration planning deals with some additional constraints coming from the underlying ISs. All the agents embed possibly existing ISs. Some of these ISs work in a synchronous manner, others in asynchronous mode. As such, at any moment, an IS may emit a request which has to be handled immediately by the distributed control. This control has also to foresee error handling, emergency and contingency planning in case of problems in the ISs.

TOWARDS ARCHON'S ARCHITECTURE

In the first paragraph we set the goals of the ARCHON system. Then we study some existing architectures to see if they provide an answer to the problem, and finally, the ARCHON architecture is presented.

1 The Goals of ARCHON

(1) Provide a cooperation framework for industrial distributed process control, embed existing IS written in different languages, running on different operating systems and ma-

chines.
(2) Give both control and data management facilities.
(3) Allow full asynchrony in input-output towards ISs, support parallel executions and the related control mechanisms.
(4) Be reactive.
(5) Be fault-tolerant.
(6) Allow an independent development of the agents

2 Existing Architectures

Blackboard architecture is one of the most inspiring problem solving paradigm in Artificial Intelligence, it addresses distributed problem solving, real-time systems and is able to handle complex data as well as their related knowledge. At a first glance, blackboard seems to be a good architecture to solve our problem.

However, even with sequential blackboards, the control needs to be improved to answer the constraints of our application, by incorporating of a blackboard for control, like in OPM, to have more opportunistic ability or flexibility at the top-level control, and to allow quasi-concurrent working of noninteracting knowledge sources and background processing of knowledge sources[2].

Implementing such an architecture in a distributed environment seem to be difficult, since experience shows that a good sequential algorithm is often not even a decent parallel algorithm. In General, three parallel and distributed Blackboard design alternatives are proposed[6]: the shared memory blackboard, the distributed blackboard and the blackboard-server approaches. The Shared memory approach is not suited to handle our problem which is geographically distributed and running on independent agents. Sharing a memory would not provide a robust architecture. The blackboard server brings communication problems with loosely coupled agents and introduces a bottleneck. The distributed blackboard is the most suited design alternative for our architecture. Implementation of such an architecture shows quite disappointing results[16] in terms of efficiency, compared to other parallel algorithms, because parallelism adds a burden a work to deal with load balancing between processors, task sharing and data handling. As those problems are still open questions, we took the following options for the design of ARCHON:
- Distributed Data and Distributed Control will be implemented through two different and independent mechanisms.
- The distributed control mechanism is reactive and fault tolerant.
- A planning and coordination module takes the charge interaction between data and control.
- The planning and coordination module performs the cooperation between agents and their load balance.

These design option are original and follow our industrial needs so to be able to use already existing software for the communication and control of the architecture, and to embed later on the OSF Distributed Computing Environment[3]. In the same manner, this design option permits to take advantage of the latest evolutions of data bases for the distributed data-processing. Finally, we hope that splitting the Distributed Data from the Distributed Control brings modularity in maintenance and evolution towards future standards.

3 ARCHON's Architecture

Figure 1 shows the ARCHON platform which is an instance of the software developed in the ARCHON project [15] to embed the Intelligent Systems. The figure focuses more precisely on the structure of an ARCHON agent: it is constituted by an IS and its related *ARCHON Layer*.

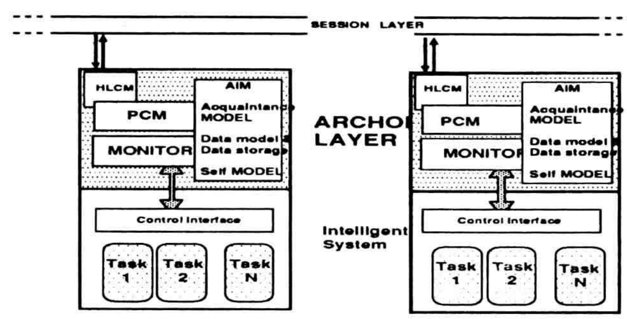

Figure 1: Architecture of ARCHON

The *Intelligent Systems (ISs)* are the systems controlled by ARCHON, they are pre-existing or built on purpose. Typically the ISs are expert systems working on a given area, or relational or object-oriented DBMSs. By extension, the *IS-Tasks* are tasks which can be carried out by an IS. ARCHON provides the mean for encapsulating applicative intelligent systems into a cooperative framework. To keep modularity and insure short response times, control and data management facilities are carried out by specific modules. An Agent encompasses four modules, one for each task to be carried out:
- *HLCM* (High Level Communication Module) All messages passing between agents is done through the HLCM which allows transparent communications.
- *PCM* (Planning and Coordination Module) The PCM provides global situation assessment, planning and supervision of all cooperating behaviour of the ARCHON community of agents. This may sometimes require high-level cooperation features. Each agent has to know its own skills as well as the ones of other agents so as to be able to share the work in the agent community. Task sharing in the agent community may be carried out in a simple and straightforward manner by imposing the work to the agents, or in a more sophisticated manner implying negotiations among the agents. The principle of contract nets is implemented for such purposes[12]. Two models are used to describe the agents: the *Agent Acquaintance Model (AAM)* represents