Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 1997 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

December 1997

Toward Design and Analysis of Organizational Intelligence through Learning Multiagent Systems

Takao Terano
The University of Tsukuba

Toward Design and Analysis of Organizational Intelligence through Learning Multiagent Systems

Takao Terano
Graduate School of Systems Management, The University of Tsukuba, Tokyo
3-29-1 Otsuka, Bunkyo-ku, Tokyo 112, Japan
Tel: +81-3-3942-6855, Fax: +81-3-3942-6829,
Email: terano@gssm.otsuka.tsukuba.ac.jp

Executive Summary

In this paper, we discuss a new direction to design and analysis problems of organizational intelligence. By organizational intelligence, we mean both intellectual behaviors of human organizations and distributed computer systems. The problems have become one of the key issues on future information management. In such complex systems like internet applications, social insects in A-Life and business process restruction processes, interesting organizational behaviors often emerge, and even very subtle control mechanisms can remarkably change the characteristics of the systems. However, we have had little knowledge on such phenomena We state the importance of synthetic approaches via computer simulation studies on real data through multiagent modeling in distributed artificial intelligence and emergent computation, especially focused on its learning functions.

First, this paper explores the theory and tools to potentially provide us to solve such design and analysis problems. Referring to the state-of-the-art literature, we discuss the feasibility of the way to integrate multiagent systems via the concepts of distributed artificial intelligence, machine learning, and genetic algorithms considering the applicability of the theories of economics and laws. Next, we propose a novel method to facilitate questionnaire data analysis using a learning multiagent system, and provide a framework for analyzing emergent behaviors of inductive learning agents. Intensive experiments on a case study of Japanese industries on corporate management in advanced-technology society show that we have got accurate and/or simple results from practical questionnaire data without explicit control mechanisms.

From these discussions, we conclude that (1) Although multiagents systems shows very complex behaviors, the design and analysis problems of multiagent systems can be resolved using the idea that the functions of the agents are essentially simple, and that simple rules are able to control emergent behaviors among the systems, and (2) Concepts from social sciences such as economics and laws could be effective to approach the problems.

1. Introduction

Design and Analysis of organizational intelligence (Blanning 1996) has become one of the key issues of future information management for both distributed computer systems and human organizations. In such complex systems, interesting organizational behaviors often emerge, and even very subtle control mechanisms can remarkably change the characteristics of the systems. This paper explores the problems how adaptive behaviors emerge in a multiagent system and then how we can design such a complex system with potentially desirable properties. We must identify common principles in such problems as how distributed knowledge systems (e.g., WWW) work, why social insects interestingly act autonomously, and what roles economical and/or law systems really have in the human society.

The objectives of the research are twofold: the first is a theoretical one to provide a framework for analyzing and designing emergent behaviors of organizational intelligence through learning multiagent systems, and the second is a practical one to develop a novel method to facilitate questionnaire data analysis. In the following sections, we describe back ground and motivation of the research, multiagent systems techniques potentially applicable to address the problems, and a proposal of learning multiagent model and its experiments in order.

2. Background and Motivation

Recently, a great deal of arguments have devoted to the study of (1) organizational intelligence with distributed information processing systems such as Internet applications, (2) organizational behaviors of animats or social insects in A-Life literature, and (3) explainable and executable models to analyze the activities of human organizations such as restructuring process analysis of corporations and principles of computer supported cooperative works.

Researches of the above categories often utilize multiagent systems and emergent computation techniques. To explore these problems, we have conducted research both on questionnaire analysis using Genetic Algorithms (GAs) and Inductive Learning (IL) (Aiba 1996), (Terano 1995) and on organizational behaviors of a multiagent system with learning functions (Terano 1996a, 1996b).

On the former topic, we have developed a system SIBILE , which acquires efficient decision rules from questionnaire data using both simulated breeding and inductive learning techniques. The basic ideas of the method are that simulated breeding is used to get the effective features from the questionnaire data and that inductive learning is used to acquire simple decision rules from the data. The simulated breeding is one of the Genetic Algorithm based techniques to subjectively or interactively evaluate the qualities of offspring generated by genetic operations (Dawkins 1986), (Unemi, 1994), (Sims 1992).

On the latter topic, we have proposed a computational model: LPC. In the model, we assume that a set of problems is given to one of the agents, any single agent cannot solve the problems, thus, the agents must communicate each other, and the agents have abilities to learn from both problem solving results and communications. The model consists of a set of agents with (a) a knowledge base for learned concepts, (b) a knowledge base for the problem solving, (c) a prolog-based inference mechanism, and (d) a set of beliefs on the reliability of the other agents. Each agent can improve its own problem solving capabilities by inductive and/or deductive learning on the given problems and by reinforcement learning on the reliability of communications among the other agents.

On the other hand, from the state-of-the-art literature on organizational behavior studies, they frequently report that small autonomous functions can generate global interesting structures and behaviors. The examples are easy to list up:

1. Some interesting organizational phenomena usually occur in observing a multiagent system with inductive learning functions and GA-based bias exchange mechanisms will give better results than the ones via single agent system (Hunter 1996), (Shaw 1996);
2. Sophisticated integration of GA-based- and conventional-machine learning mechanisms will be effective in (parallel) problem solving (Fogarty 1995), (Wilcox 1995);
3. Several simulation studies reveal that organizational problem solving will be improved via fancy ad-hoc techniques (Carley 1994).
4. Knowledge intensive learning in a multiagent system will generate interesting results (Aiba 1996), (Terano 1995).

These researches in the literature only discuss the emergent properties which can be identified from the observers' standpoints, and tend to ignore the design problems of such multiagent systems. These researches have not yet attained to describe the flexibility and practicability of real organizations. By the word flexibility and practicability, we respectively mean that ambiguous knowledge sharing and ongoing adaptation in the context of activities occur.

We believe that it is critical to have methods both to observe the emergent phenomena in multiagent systems and to give desirable properties to them. The key to succeed in the design and analysis of organizational behaviors is to concentrate on (1) the characteristics of generate-and-test functions of the systems, and (2) the functions of knowledge intensive problem solving, communicating and learning, (3) the characteristics of problems the multiagent system solve, and (4) institutional issues of organizations such as cooperative activities, marketing auctions, and common laws (Epstein 1996), (Sowell 1991). We state the importance to introduce the concepts of social sciences into the design of effective multiagent systems.

3. Problem and Agent Specifications of a Multiagent System

Problems given to a multiagent system and corresponding roles of the agents usually have various kinds of parameters. In the following, some of them we consider important are listed.

1. Design of the agent: The abilities of the agents in a multiagent system can be heterogeneous or homogeneous. That is, the agents may or may not have the same kinds of problem solving methods and knowledge structures. If the agents are heterogeneous, we would design and simulate complex activities of human organizations, however, the analyses would become difficult. If they are homogeneous, on the other hand, the analyses would become easier, however, we can only make models of simple organization as described various A-life and artificial society papers.

2. Design of organizational structures: Organizational structures of a multiagent systems may be pre-determined, changing through problem solving processes, or emerging among the agent interactions. In each case, we must determine appropriate parameter values and functions of the agents.

3. Interaction with environments: We must define the multiagent system is open or not against the environments. This is one of the characteristics of emergent phenomena, that is, if we assume there exist observers who investigate emergent phenomena would occur from the outside of the environment, parameters to determine the interaction with the environments could be controllable, if not, the multiagent system could autonomously shows its own behaviors.

4. Task characteristics: Tasks with examples and/or problems which are given to a multiagent system must be carefully designed. If only one of the agents are given the problem, it will be distributed to the others. When the problems are simultaneously given to the system, if the examples are different from each other, it will also cause various problems.

As described above, the characteristics of a multiagent system may be very sensitive against its parameter changes. Therefore, if we determine the critical parameters, we could easily design and control the structures and behaviors of the whole systems. Simple rules can control the structures and behaviors. Institutional structures such as corporations and marketing auctions and corresponding legal or economical systems could play important roles in designing a multiagent systems.

For example, the rationality of human systems characterized by cost-benefit trade-offs are categorized into economic, social, and political trade-offs discussed by (Sowell 1980). However, from the discussions of (Epstein 1995), complex law systems which prescribe social trade-offs in human societies can be described by the four simple rules to explain: the roles of individual autonomy, the acquisition of private property, the transfer of human and material resources under a regime of private contract, and the protection of persons and property from aggression afforded by the tort law. These concepts themselves are come from social sciences, and, therefore, cannot be operationalize in the sense of computer sciences, however, based on these concepts, it can be possible to design and control the organizational behaviors of complex multiagent systems.

4. Experiments of a Learning Multiagent System

In the following, we describe the intermediate results of the research: how we build a multiagent system with learning mechanisms. The multiagent system in this section consists of inductive learning agents with ID3 type concept learning programs (Quinlan 1986), and solves feature selection problems. Feature selection is the problem in machine learning and statistical researches to choose a small subset of features that is necessary and sufficient to describe the target concept (Weiss 1991). The system organizationally and iteratively learns good descriptions of the target concepts via a Genetic Algorithm (GA)-based feature exchange mechanism.

Our hypotheses are: (1) A multiagent system with inductive learning functions and GA-based bias exchange mechanisms will give better results than the ones via a single agent system and will be practical in parallel problem solving (Hunter 1996), (Shaw 1996); and (2) Organizational learning or problem solving functions will evolve in such a multiagent system (Blanning 1996), (Carley 1994)

4.1. Algorithm for Feature Selection in a Multiagent System

Each agent of the proposed method consists of a (sub-)set of attributes and C4.5 inductive learning program. Common training examples or questionnaire respondents data are given to the system. However, the background knowledge or bias of the agents are different, that is, each agent only know a subset of features in the training examples.

The agent has its own features or questions in the questionnaire, and executes inductive learning only using the attributes. If some of attributes are meaningless, the outputs of C4.5 program, i.e., decision trees, do not contain the meaningless attributes. Therefore, the attributes appeared in the decision trees are considered to be effective ones. Please note that only such a subtle control mechanism has a role of selection schemes based on (implicit) objective functions and that without explicit comparisons of objective functions some good results will be able to evolve as is found in natural selection and breeding of living things.

The effective features are collected into a pool, which corresponds to the pool of memes in (Hunter 1996). The recombined set of features are selected from the pool. We call the recombination scheme as Organizational Crossover, because the offspring are generated from all the individuals in the system instead of two parents in usual GA operations.

The following procedure is considered as a modification of conventional GA-based feature selection (e.g., (Bala 1995)) in such ways that (1) we adopt Organizational Crossover operator and (2) we do not apply usual selection methods such as fitness scaling, and so on.

Step 1: Initialization
Generate the initial population, that is, we select m sets of individuals with less than or equal to l features. The m and l respectively represent the number of individuals and the length of their chromosomes.

Step 2: Apply Inductive Learning
Based on the information of each of m agents with the selected features suggested as '1' in the chromosome, the data acquired from the questionnaire is aggregated, each of which has the corresponding features in it. Then the m sets of the data are processed by inductive learning programs. As a result, m sets of decision trees with selected features or the corresponding set of decision rules are generated.

Step 3: Organizational Crossover

(1) Feature Collection The features appeared in the decision tree of the agents are collected into a bag, which corresponds to a meme pool to apply crossover operations. Then add one of the features is randomly to the pool to mutate the memes.

(2) Offspring Generation From the pool, a new m set of features are selected to each agent in order to get new sets of offspring. The elitist individual remains unchanged.

Step 4: Repeat the Steps
Steps 2 to 4 are repeated until an appropriate decision tree or set of decision rules is obtained.

4.2. Experiments

To validate the effectiveness of the proposed method, we have carried out experiments from a practical case study on the questionnaire to Japanese industries on corporate management in advanced-technology society (UA 1995) meda. The data is used to classify major 138 respondent companies into two classes based on the 21 continuous features in the questionnaire.

4.2.1. Preliminary Experiments on a Single Agent

To compare the results in the following subsections, we have carried out the application of C4.5 programs to a single agent, varying parameters of C4.5. Such experiments are very usual to apply inductive learning program to practical data. The results are summarized as follows:

(1) Using defaults parameters:
- Number of Attributed in the Decision Tree: 11
- Number of Decision Rules Generated: 5
- Correctness of the Rules Applied to Training Data: 74 %

(2) Tuning Parameters to Simplify the Rules:
- Number of Attributed in the Decision Tree: 2
- Number of Decision Rules Generated: 2
- Correctness of the Rules Applied to Training Data: 30 %

The results can not be used for decision making, because the correctness is too low.

(3) Parameter Tuning to Improve the Correctness:
- Number of Attributed in the Decision Tree: 11
- Number of Decision Rules Generated: 14
- Correctness of the Rules Applied to Training Data: 76%

The results also cannot be used for decision making, because the resulting rules are too complex to interpret.

The correctness of the rules with default parameters is a usual one applied C4.5 to questionnaire data. Tuning the parameter of C4.5, we are able to simplify or improve the decision rules. However, the results are not good and the tuning task is very hard even for human experts.

4.2.2. Experiment with Conventional GA Techniques

We apply conventional GA-based feature selection with uniform crossover operation for the objective functions: Case 1:Minimize the number of attributes in the decision tree, and Case 2: Maximize the correctness of the decision tree. The number of individuals are set to 10 in each experiments. (The number seems to be too small, however, it is usual in the experiments of simulated breeding methods with human-in-a-loop interactions. In the future, we would like to compare the experiments of the paper with the ones with which human experts is concerned(Terano 1996a)

The parameters of C4.5 programs are set to default ones. The process converges after 10 to 20 iterations with 10 different epocs. The results are summarized as follows:

(1) Minimization of the Attributes:
- Number of Attributed in the Decision Tree: 8-13
- Number of Decision Rules Generated: 5-15
- Correctness of the Rules Applied to Training Data: 20%-40%

(2) Maximization of the Correctness:
- Number of Attributed in the Decision Tree: 10-15
- Number of Decision Rules Generated: 8-15
- Correctness of the Rules Applied to Training Data: 30%-50%

These results suggest that usual GA-based techniques does not work well in the application domain.

4.2.3. OC-Based Feature Selection

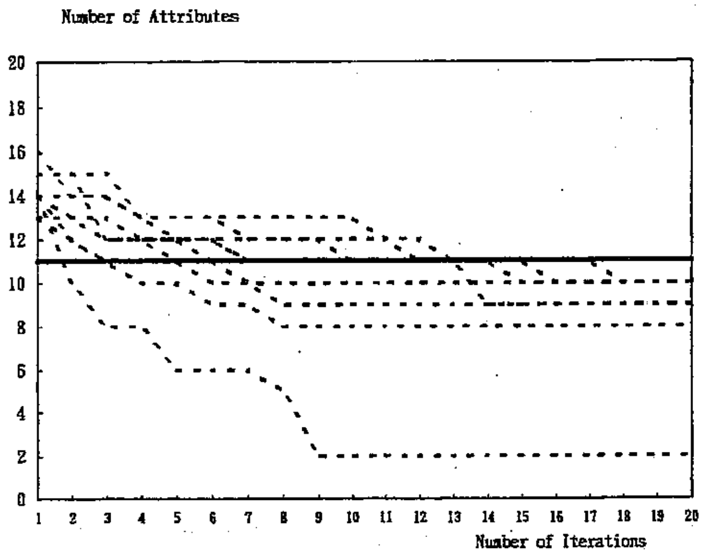

Figure 1 Effects of OC-Based Feature Selection to Minimize the Number of Attributes

We apply the proposed method to the same data. The parameters of C4.5 programs are set to default ones. In the experiments, we tries to find Case 3: Decision trees with small number of attributes and Case 4: Decision trees with high correctness. To extract these, we only adopt the elitist strategy and do not utilize selection schemes based on the values of the objective functions. By the elitist strategy, we mean the individual with smallest number of attributes in Case 3 and the highest correctness in Case 4 remains in the next step. The experiments are carried out on 10 epocs in each case.

(1) Minimization of the Attributes:
- Number of Attributed in the Decision Tree: 2-10
- Number of Decision Rules Generated: 3-10
- Correctness of the Rules Applied to Training Data: 50%-70%

The relation of the number of iteration and the changes of number of the attributes in the 10 epochs are shown in Figure 1. The straight line shows the number of attributes generated by a single agent with C4.5 best parameters. The dotted lines correspond to the GA-based iteration in each epoch. As is shown in Figure 1, the number of attributes converges smaller values than the one of the cases of a single program and conventional GA-based methods. In the best case, the number of rules is only three and the rules contain only two attributes. However, the correctness is acceptable in the application domain.

(2) Maximization of the Correctness:
- Number of Attributed in the Decision Tree: 8-12
- Number of Decision Rules Generated: 8-12
- Correctness of the Rules Applied to Training Data: 78%-81%

The relation of the number of iteration and the changes of the correctness in the 10 epochs are shown in Figure 2. The straight line shows the correctness of rules generated by a single agent with C4.5 best parameters. The dotted lines correspond to the GA-based iteration in each epoch. As is shown in Figure 2, the correctness also converges smaller values than the one of the cases of a single program and conventional GA-based methods. Also, the size of the decision trees is feasible to interpret by human experts.

Based on the experiments, several discussions are given. As is shown above, the idea of Organizational Crossover or the pool of memes seem promising to solve feature selection problems in a learning multiagent system. Although the procedure is very simple, it works well, when we apply it to a practical data analysis problem, which is characterized by very-noisy, so-many-attributes, and so-many-correlated data.

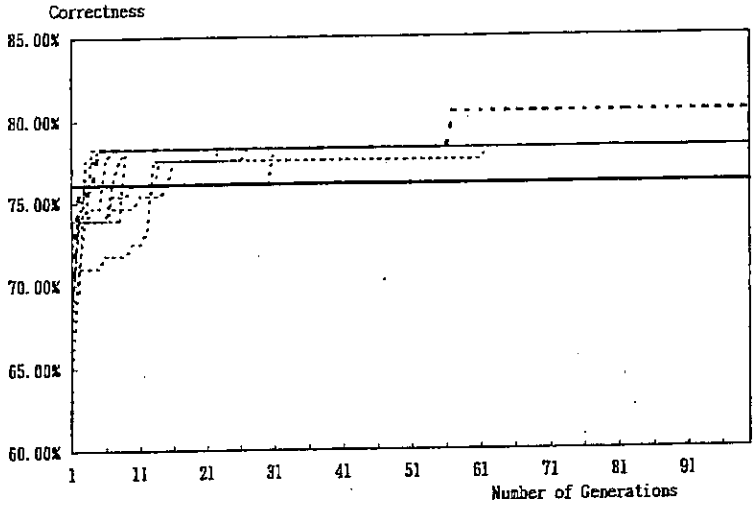

Figure 2 Effects of OC-Based Feature Selection to Maximize the Correctness

The most important implication of the experiments is that we may be able to control the emergent phenomena in a multiagent system by changing parameters of the system. Using the above concepts, we will be able to develop vehicles to design, analyze and control emergent behaviors or group activities in distributed systems, which are often observed in organizationally intelligent activities such as WWW and/or groupware systems with CSCW functions.

5. Concluding Remarks

This article has addressed the problems to design and analyze complex multiagent systems, which will consist of autonomous agents with some problem solving and communication functions. There are so many multiagent systems available, e.g., distributed computer systems such as WWW in Internet, biological systems of social insects, and various organizational intelligence research of corporations in human societies, however, we have few basic theories for them. Referring to the state-of-the-art literature, we have discussed the feasibility of the way to integrate multiagent systems via the concepts of distributed artificial intelligence, machine learning, and genetic algorithms considering the applicability of the theories of economics and laws. We have also proposed a novel method with learning multiagent systems applicable to practical data analysis problems.

Our intermediate conclusions are simple: (1) Although multiagents systems shows very complex behaviors, the design and analysis problems of multiagent systems can be resolved using the idea that the functions of the agents are essentially simple, and that simple rules are able to control emergent behaviors among the systems, and (2) Concepts from social sciences such as economics and law could be effective to approach such problems. The research has just begun, therefore, so many problems would remain, however, we believe that the field will be fruitful.