An Immune Optimization Inspired by Biological Immune Cell-cooperation for Division-and-labor Problem

Naruaki Toma
Graduate Student,
Doctral Couse in Complex Intelligent, Systems Engineering, Graduate School of Science and Engineering, University of the Ryukyus, JAPAN.
tnal@eva.ie.u-ryukyu.ac.jp

Satoshi Endo, Koji Yamada, Hayao Miyagi
Department of Information Engineering, Faculty of Engineering, University of the Ryukyus, JAPAN.
{endo,koji,miyagi}@ie.u-ryukyu.ac.jp

Abstract

The purposes of this paper are to propose and evaluate an immune optimization algorithm inspired by biological immune cell-cooperation, and this algorithm solves the division-of-labor problems in multi-agent system (MAS). The proposed algorithm solves the problem through interactions between agents, and between agents and environment. The interactions is performed by division-and-integration processing inspired by immune cell-cooperation and like co-evolutionary approach. The division-and-integration processing optimizes the work domain, and the like co-evolutionary approach performs equal divisions. To investigate the validity, this algorithm is applied to "N-th agent's Travelling Salesmen Problem" as a typical problem of MAS. The good property on solving for MAS will be clarified by some simulations.

1. Introduction

Adaptive problem solving techniques, such as neural networks and genetic algorithms, are based on information processing in biological organisms and are applied on many kinds of optimization problems. A biological immune system is one of the adaptive systems and the studies are making advances[1, 2, 3, 4]. The biological immune system is widely recognized as one of the adaptive biological system whose functions are to identify and to eliminate foreign materials.

In this paper, we propose and evaluate an immune optimization algorithm inspired by biological immune cell-cooperation, and this algorithm solves the division-of-labor problems in multi-agent system (MAS). For the following reason, it is very useful to solve these problems for application. As the reason, we need to correspond to problem space size that becomes more complicated in recent years. MAS is a study in the field of distributed artificial intelligence and attracts attention as a framework for solving effectively using cooperations or competitions through interactions. Therefore, it is very meaningful to examine the application to MAS.

The proposed algorithm solves the problem through interactions between agents, and between agents and environment[7]. This method for implementation of the interactions uses *division-and-integration processing* inspired by *immune cell-cooperation*. There are three functions in our algorithm: division-and-integration processing and *like co-evolutionary approach*. The division-and-integration processing optimizes the work domain, and the like co-evolutionary approach performs equal divisions. Then, in order to investigate the validity of the proposed method, this algorithm is applied to "N-th agent's Travelling Salesmen Problem (called n-TSP)" as a typical problem of multi-agent system. The good property on solving for MAS will be clarified by some simulations.

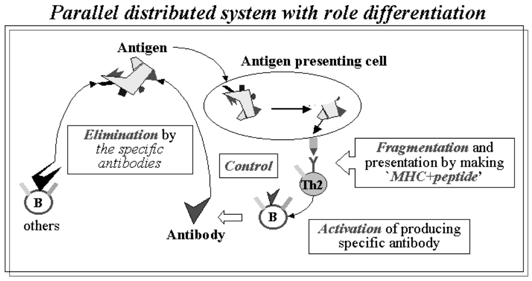

Figure 1. Concept of immune cell-cooperation.

2. Analogy from biological immune system

In the biological immune system, the following two mechanisms are considered important in eliminating invaded antigens[5]. First, *major histocompatibility complex* (MHC) is used to distinguish a "self" from the other "nonself" when nonself invades self. Second, *immune network* is composed of immune cells and their connections, and then specific antibodies are produced by modification of the cells. Then, a framework included the two functions which eliminate of unknown vast antigens in parallel, is called *'immune cell-cooperation'*. In the point of view of engineering system, it is considered the immune cell-cooperation to be a parallel distributed system with role differentiation. The roles in this system are (1) fragmentation and presentation of antigens, (2) activation of producing specific antibodies, (3) elimination of the antigens by specific antibodies, and (4) control of the functions (see figure 1). We have as object to model, implement and apply to MAS. Especially, we choose the fragmentation, the activation and the elimination functions in this issue because their functions are minimum components to perform as problem-solvers.

We construct an optimization algorithm based on a concept that (1) fragments a problem, (2) solves the fragmented sub-problems by the specific sub-solutions, and then (3) solves whole the problem through combination of these sub-solutions. We call these procedures *division-and-integration processing*.

In addition, we introduce a co-evolutionary concept that the immune system evolves against the virus which evolves so as to escape the elimination of the system. We call the procedure *like co-evolutionary approach*. In order to apply this concept as a searching method in MAS, we construct a procedure liken the escaping virus to changing environments.

3. Issues of division-of-labor problems

MAS that is a study in a field of distributed artificial intelligence is an information processing technique that autonomous agents solves problems by interactions among their agents. The key subject of division-of-labor problems is to focus on the issues of distribution of works for agents. The solvers of the problems are defined as follows: the domain that each agent covers is defined as work domains (WD), and the aggregate total of the work domains is defined as problem domain (PD). There are two aims in the division-of-labor problems. They are: (a) each WD_i that is assigned by $agent_i$ must be divided evenly and, (b) the system performs effective division-of-labor by optimizing each WD_i.

As a typical case problem of the division-of-labor, we deal with n-TSP to investigate an adaptation ability of our model. Travelling salesman problem (TSP) is one of the most typical combinatorial optimization problems. The objective is to find a roundtrip of minimal total length visiting each node exactly once. The number of salesman in n-TSP was extended from singular to plural number. The objective is finding the minimal tour by division among salesman. Note that one condition that all salesmen use the same city as the starting point was set in our experiments.

4. Proposed method

The algorithm solves the problems through two searching ways, (1) *division-and-integration processing* by agents and (2) *like co-evolutionary procedure* by components in the environment. The procedures of the extended algorithm against a n-TSP are described as below. In the procedures of proposed method, Step 1 is processed by the System for initialization once for all. Each agent processes repeated from Step 2 to Step 6 for optimizing the problem.

[Step1. Definition of problems and immune functions.]
Cities and Salesmen as the problem must be defined. Salesman's unique ID and division-and-integration processing for tours of each salesman must be defined. At this point, each city has the ID for expression of salesman visiting the city.

[Step2. Calculation of objective function.]
The cost of salesman is calculated by following function.

$$Cost(S_i) = \sum distance(tour_{S_i}) \tag{1}$$

- S_i: Salesman i.
- $tour_{S_i}$: the tour of Salesman i.

[Step3. ***division processing*** **with Simulated Annealing.]**
One salesman tries to divide own tour into two subtours for searching lower costs according to these steps (see also figure 2-(A)). First, decide of any one $subtour_j$ for new other $salesman_j$. Second, make new $tour_i$ of $salesman_i$ except the $subtour_j$. Third, calculate cost of both new sub-tours. If the cost, after division, is improved, the division will be performed. As a consequence, in the case of DIVISION performed, the original salesman generates of a new agent for making efficient results.

[Step4. ***integration processing.*****]**
Two salesmen try to integrate the tours for searching lower costs by following steps (see also figure 2-(B)). First, decide of any two tours and integration point ptr. Second, make new tour by binding at the ptr. Third, calculate cost of new tour. If the cost, after integration, is improved, the integration will be performed. As a consequence, in the case of INTEGRATION performed, the original salesman kills the other salesman for making efficient result.

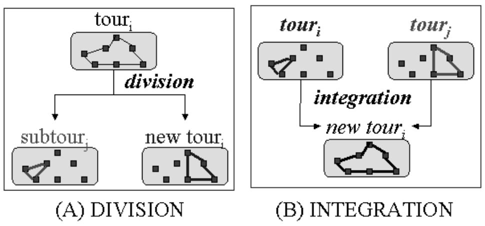

Figure 2. Examples of Division and Integration.

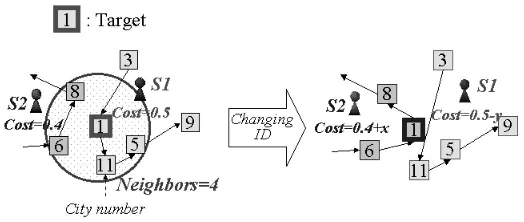

Figure 3. Like co-evolutionary approach.

Table 1. Distribution of the number of searching solutions.

	Trials	Times of execution
DIVISION	6,242	221 (6)
INTEGRATION	546	225 (16)
Mutation	6,037	50
CITY operation	47,876	47,876
Total	60,701	48,372

[Step5. Mutation.]
Swap any two cities. If the cost, after swap, is improved, the swap will be performed.

[Step6. Like Co-evolutionary Procedure.]
This approach changes ID of a city depending on neighbors cost to process following steps (see also figure 3). First, check costs of salesmen that visited neighbor cities of target city. The neighbors is defined as N-cities close to the target city. In this example, the neighbors are visited by $S1$ and $S2$. Second, if the other salesman's cost lower than its cost, the city changes its ID into the other. As a consequence, in this example, the ID of target city is changed to $S2$, and then, the target city is visited by $S2$ like the right side of figure 3.

In the manners, each agent searches for lower costs, and each component in the environment searches for even divisions. Note that, in order to push up the number of execution-times of division-processing, 'Simulated Annealing' is implemented at step 3.

5. Experiment

5.1. Design of experiment

In order to investigate the basic performance of our method, we apply this algorithm to the following n-TSP. Definition of problem is set in table 4. The parameters of the extended algorithm are shown in table 5.

Figure 4. Definition of problem.

the number of cities	25
the number of salesmen	3
the arrangement of cities	dual circle
start city	center of circle
radius (Out,In)	0.4, 0.384
terminal steps	2000

Figure 5. Parameters of the algorithm.

the number of neighbors	4
the number of initial agents	under 24
terminal steps	2000

5.2. Results and discussions

Figure 6 shows an acquired solution. As a result of the extension of our method, a best solution that has equal divisions and optimized cost. The transition of the number of visited cities for each agent is shown in table 1. And the number of times of execution of division-processing improved owing to the Simulated Annealing. As a result, the integration-processing is performed frequently and effectively, also.

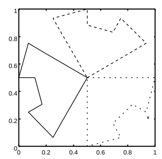

Figure 6. An obtained result.

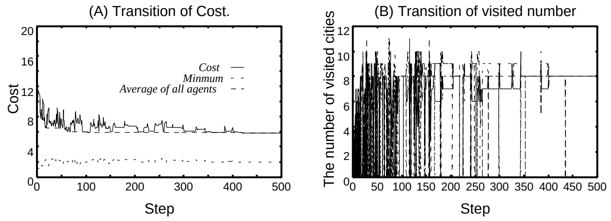

Figure 7. Transition of cost and the number of visited cities.

In order to verify the behaviors of our method, the transition of the cost, sum of whole agent's cost, and the number of visited cities (dividing-number) are shown in figure 7. The cost improves dramatically up to step 50. And an appearance of dividing changes from a combination of a few number of visited cities with great number of agents into a combination of approximately eight cities with 3-5 agents. The appearance is caused under the primary influence of the equalizing divisions in step 6 of the extended algorithm. Then, a rise of average cost of all agents occurs also. In subsequent steps, the optimization is performed repeatedly while maintaining the approximately even divisions. Finally, the transition of fitness and dividing-number converges on the optimum solution that is a combination of eight cities with three agents in figure 6.

6. Conclusion

In this paper, we proposed and evaluated an immune optimization algorithms in order to verify the engineering application possibility of artificial immune system. Our method solves the problems through combination of division, integration and co-evolutionary approach. These functions are based on local interactions between agents, and between agents and environment. In MAS, clarifying the objective function considered all agents and components in the environment is too hard problem, and then it is important optimizing of the whole problem by using the local interactions. Since our algorithm can optimize division-of-labor problems, it can expect what is functioned effectively as an optimization algorithm in MAS.

The first author acknowledges the Grant-in-Aid for Scientific Research (Grant-in-Aid for JSPS Fellows).

References

[1] H.Bersini and F.J.Varela (1991). The Immune Recruitment Mechanism : A Selective Evolutionary Strategy. Proc. of ICGA 91.

[2] J.D. Farmer, N.H. Packard, A.A. Perelson (1986). The Immune system adaptation, and machine learning. Physica 22D, pp.187-204,1986.

[3] S. Forrest, A.A. Perelson (1990) Genetic algorithm and the Immune system. Proc. of 1st Workshop on Parallel Problem Solving from Nature, Dortmund, Federal Republic of Germany, 1-3, October, 1990.

[4] Y.Ishida and N.Adachi (1996). Active Noise Control by an Immune Algorithm : Adaptation in Immune System as an Evolution. Pro. ICEC 96, pp.150-153.

[5] Charles A. Janeway, Jr., Paul Travers ; with assistance of Simon Hunt, Mark Walport (1997) Immunobiology : The Immune System in Health And Disease. Garland Pub.

[6] N. Toma, S. Endo, K. Yamada (2000). The Proposal and Evaluation of an Adaptive Memorizing Immune Algorithm with Two Memory Mechanisms. Journal of JSAI, Vol.15, No.6, pp.1097-1106.

[7] N. Toma, S. Endo, K. Yamada, H. Miyagi (2000). The Immune Distributed Competitive Problem Solver with MHC and Immune Network. Proceeding of ANNIE 2000, Volume 10, pp.317-322.