A PROPOSAL OF AN IMMUNE OPTIMIZATION FOR DIVISION-OF-LABOR PROBLEMS USING IMMUNE CELL-COOPERATION AND TOLERANCE

NARUAKI TOMA, SATOSHI ENDO, KOJI YAMADA, HAYAO MIYAGI

Department of Information Engineering, Faculty of Engineering,
University of the Ryukyus, Okinawa 903-0213, Japan
Tel: +81-98-895-8714, Fax: +81-98-895-8727
E-mail: tnal@eva.ie.u-ryukyu.ac.jp, {endo,koji,miyagi}@ie.u-ryukyu.ac.jp

Abstract

The purpose of this paper is to propose an immune optimization algorithm using division as well as integration processing based on immune cell-cooperation and to investigate its validity by computer simulations. In the biological immune system, the immune tolerance is a phenomenon which an eliminating behavior against the invading antigens doesn't workable (tolerate) under two special conditions, in cases that the amount of antigens is too low or too high. We use the immune tolerance to decide a well-suited operation point for acquirement of even division of work domains. Second, the immune cell-cooperation is a framework including MHC and immune network, the function of which is to eliminate unknown vast antigens. Our algorithm solves the division-of-labor problems for each agent's work domain inside the multi-agent system (MAS) through interactions between two agents, and those of between agents and environment through the work of immune functions. There are three functions in our algorithm: the division as well as integration processing and the co-evolutionary-like approach. The division as well as integration processing optimizes the work domain, the co-evolutionary approach realizes equal divisions, and the tolerance function helps the optimizing and equalizing. In order to investigate the validity of the proposed method, this algorithm is applied to the "N-th agent's Travelling Salesmen Problem (called the n-TSP)" as a typical problem of multi-agent system. The property that is believed to function as solution driver for MAS shall be clarified using a simulation.

Keywords: optimization, immune tolerance, immune cell-cooperation, division-of-labor problem

1 INTRODUCTION

Adaptive problem solving techniques, such as neural networks and genetic algorithms, are based on information processing in biological organisms and are applied in many kinds of optimization problems, see [2, 4]. A biological immune system is one of the adaptive systems and the studies of the said system are making progress these days, see [5, 3]. In the biological immune system, the following mechanisms are considered as important functions in eliminating invaded antigens [1], which mechanisms are called immune tolerance, cell-cooperation, and co-evolutionary phenomenon against the antigens. First, the immune tolerance is a phenomenon which an eliminating behavior against the invading antigens doesn't workable (tolerate) under two special conditions, in cases that the amount of antigens is too low or too high. We use the immune tolerance to decide a well-suited operation point for acquirement of even division of work domains. Second, the immune cell-cooperation is a framework, the function of which is to eliminate unknown vast antigens by co-operations among some kinds of immune cells in parallel. Especially, the function with role differentiation against specific antigens is considered as important characteristic for parallel-distributed system. The 'splicing' is one of the re-combination operator of genes, which function is used for forming the role. The division as well as integration processing in our method are based on the splicing. And then, third, the co-evolutionary phenomenon that the invading antigens is evolving in competition with the immune system by some escaping methods. In our method, we use the co-evolutionary phenomenon to search for common ground for even division of work domains.

In this paper, we propose an immune optimization algorithm for division-of-labor problems using the immune tolerance, the cell-cooperation and the co-evolutionary phenomenon. Our algorithm solves the division-of-labor problems for each agent's work domain in multi-agent system (MAS). The immune tolerance is used to decide a well-suited operation point for acquirement of even division of work domains. The immune cell-cooperation is used to search an optimum work domains through a performing roles. And the co-evolutionary-phenomenon is used to search for common ground for even division of work domains. By introducing such models, we construct an adaptive optimization algorithm to solve division-of-labor problems. Then, we apply the proposed algorithm to n-th agent's travelling salesman problem (called n-TSP) which is typical problem in MAS. Some computer simulations are designed to clarify both a basic performance and characteristics featuers of proposed immune algorithm.

2 MAS AND DIVISION-OF-LABOR PROBLEMS

2.1 Division-of labor problems

Multi-agent system, which is a study in the field of distributed artificial intelligence, is an information processing technique in which autonomous agents solve problems by interactions among the agents. The key subject of division-of-labor problems is to focus on the issues of distribution of work for agents. The solvers of the problems are defined as follows: the domain that each agent covers is defined as work domain (WD), and the aggregate total of the work domains is defined as problem domain (PD). There are two issues to 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 work operation in each WD_i (see Figure 1).

2.2 N-th agent's TSP

As a typical case problem of the division-of-labor, we dealt with the nth agent's Travelling Salesman Problem to investigate the adaptability of our model. Travelling salesman problem (TSP) is one of the most typical combinatorial optimization problems. The objective is to find a minimal roundtrip route visiting each node exactly once. The number of salesmen in the n-TSP was increased from singular to plural number. The objective is to find the minimal tour route by division among salesmen (see Figure 2). Note that one condition was set (All salesmen use the same city as the starting point).

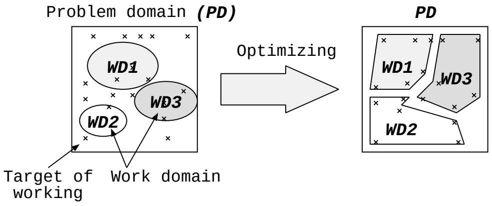


Figure 1: Division-of-labor problems optimization.

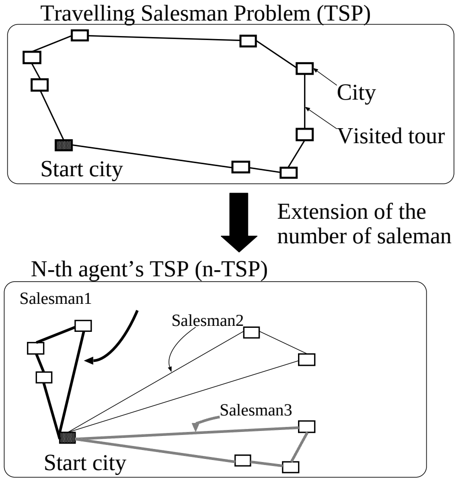


Figure 2: Illustration of n-TSP.

3 PROPOSED METHOD

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

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

[Step2: Calculation of compressed point.] Each salesman have a string expression as sub-solutions, which the sub-solution is sub-tours in n-TSP. Any substrings (consist of 'City') in the string expression are compressed when the density is too low or too high, which the density is calculated by equation 1. The compressed cities are dealt with a new city in the following procedures. By repeating this process, all of the density for the city are equal.

$$Density(City_i) = \frac{1}{\frac{dist(City_i,City_j)+dist(City_j,City_k)+dist(City_k,City_i)}{dist(ALL)}} \tag{1}$$

- $dist(City_i, City_j)$: The distance between $City_i$ and $City_j$.
- $dist(ALL)$: The total distance for the salesman (equal to equation 2).

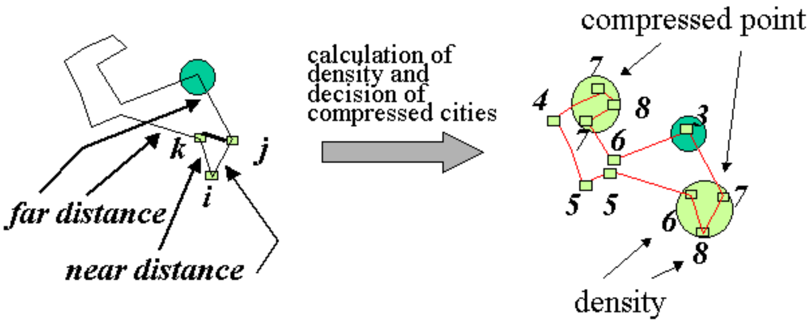


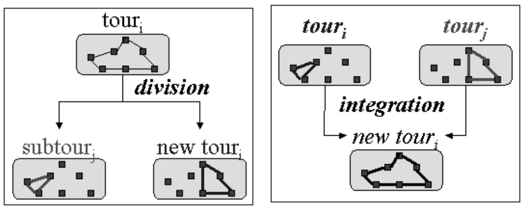
Figure 3: Illustration of compression processing.

[Step3: Calculation of objective function.] The cost of salesman is calculated by following function.

$$Cost(S_i) = \sum distance(tour_{S_i}) \tag{2}$$

- S_i: $Salesman_i$.
- $tour_{S_i}$: A visited tour of $Salesman_i$.

[Step4: Division processing with Simulated Annealing.] One salesman tries to divide his own tour into two sub tours in search of lower costs according to these steps (see also Figure 4-(A)). First, we need to decide on one sub $tour_j$ for new $salesman_j$. Second, we need to map a new $tour_i$ for $salesman_i$ except for the sub $tour_j$ and third, we need to calculate the cost of both new sub tours. When the cost after the division is improved, then the division shall be carried out. Therefore, in the case of DIVISION performed, the original salesman generates a new agent for attaining results that are more efficient.



(A) DIVISION (B) INTEGRATION

Figure 4: Examples of Division and Integration.

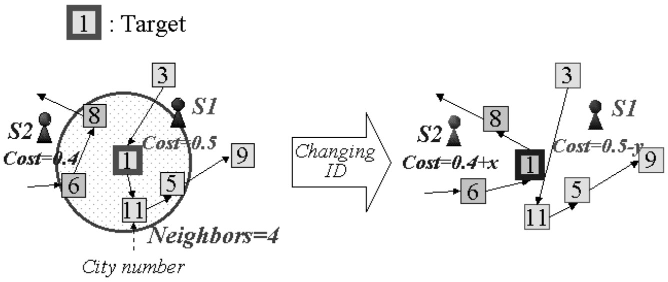


Figure 5: Escape processing.

[Step5: Integration processing.] The two salesmen try to integrate the tours in search of lower costs by following the steps (see also Figure 4-(B)). First, we need to decide on any two tours and integration point ptr has to be made. Second, routing of new tour crossing the ptr has to be scheduled. Third, the costs of the new tour have to be computed. If the cost, after integration, is improved, the integration will be carried out. Therefore, in the case of INTEGRATION performed, the original salesman kills the other salesman for making an efficient result.

[Step6: Mutation.] Swap any two cities. In the event the costs after swapping are improved, the swap will be carried out.

[Step7: Calculation of objective function.] The cost of city is calculated by the same function in case of salesman. In other words, the cost of city is equal to the visiting the city salemsman's cost.

[Step8: Escape Processing.] This approach changes the ID of a given city depending on the costs to and from the neighboring city and the costs to process the following steps (see also Figure 15). First, check the costs of salesmen that visited neighboring cities of the target city. The neighbors are 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 is lower, then the city changes its ID into that of the other. Consequently, in this example, the ID of the target city is changed to S2, and then, the target city is visited by S2 as shown in the right side of Figure 15.

[Step9: Return to the uncompressed states for the compressed points.] The compressed cities by the compression in step2 return to the uncompressed states.

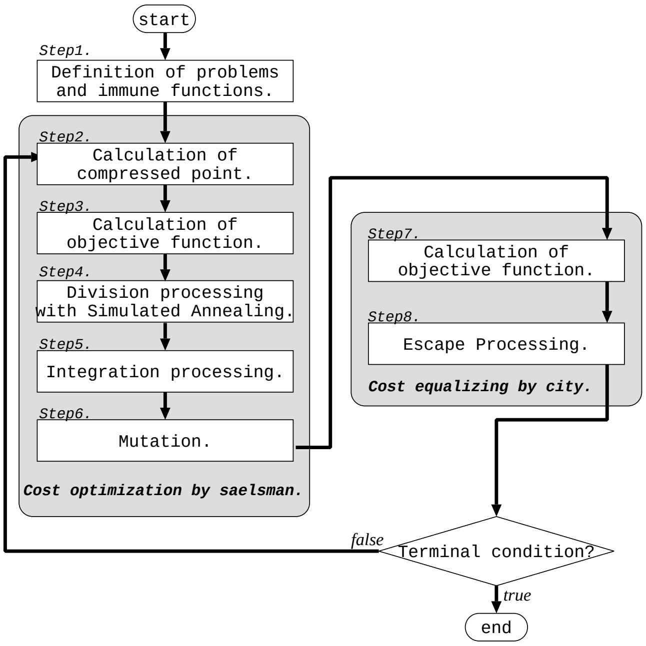


Figure 6: Procedures of proposed algorithm.

In this manner, each agent searches for lower costs, and each component in the environment searches for even divisions. Note that, in order to increase the number of execution times of division processing, 'Simulated Annealing' is implemented at Step 3.

4 EXPERIMENT

4.1 Definition of simulation

To confirm the basic performance of our method, we apply the method to one example of TSP that is publishing on TSPLIB. TSPLIB is a library of sample instances for the TSP from various sources and of various types [1] . We adopted *eli51.tsp* as one example for a simulation and the best solution in case of one salesman illustrated to Figure 7. In this simulation, we'll try to apply our method to the *eil51.tsp* under that the smallest number of salesman set *three*. Other parameters are shown in Table 1.

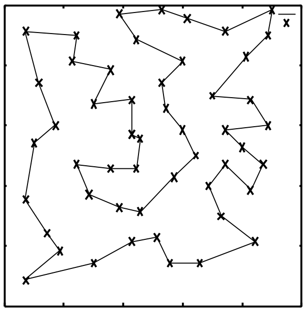


Figure 7: Best solution with one salesman for *eil51.tsp*.

---

[1] http://iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/

Table 1: Parameters.

Threshold for compression	60% lower/higher than ave-density.
The number of neighbors	4
The number of initial agents	20
Terminal steps	500

4.2 Results and discussions

Figure 8 shows an acquired solution. The solution is quite good because some parts of the tours similar to best solution with one salesman.

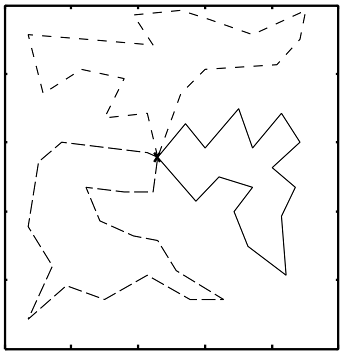


Figure 8: Result for dividing with three salesmen.

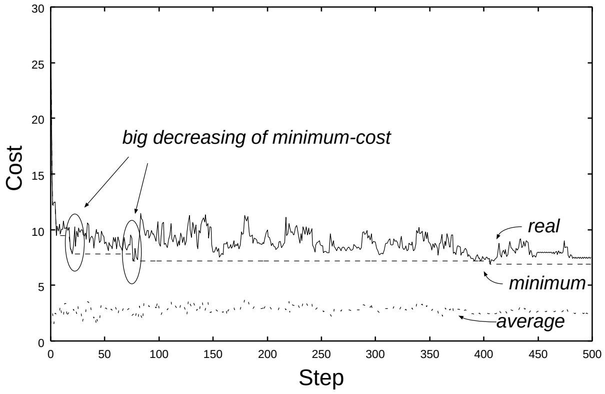


Figure 9: Transition of cost (with average,minimum).

Next, to verify the behaviors of the method, the transition of the cost shows figure 9. In the vicinity of step20 and step75, the big decreasing of cost occurred.

The modification in step75 shows in Figure 10. The left side of Figure 10 illustrates solution in step75 and the compressed cities (in this step, only two cities are compressed). The right side illustrates solution in step75 and the sub-solution (five cities in a circle) in this neighborhood of the compressed cities are modifying to get into a

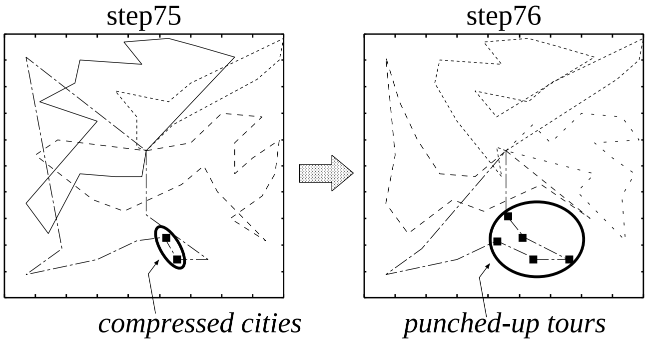


Figure 10: Modifying a tour from step75 to step76 with compression.

part of obtained solution (Figure 8). The compression occurs in case that the density of city is too low or too than average. So, the behavior depends the threshold and a formulation of the density. We'd like to investigate the affect, in the future works

5 CONCLUSION

We proposed the immune optimization using the immune tolerance, cell-cooperation and co-evolutionary phenomenon for division-of-labor problem. On basis these immune functions, our method consists of (1) the compression processing for the decision of dividing work domains, (2) the division as well as integration processing for optimizing the work domain, and (3) the escape processing realizes equal divisions. This method was applied to one n-TSP, and validity was examined. In the experiments, the property that the work domain of each agent is moved with keeping the fitness was confirmed. And, it aspects that our method is good performance. As future works, we're going to investigate the affect of the compression processing, and compare with other methods. The first author acknowledges the Grant-in-Aid for Scientific Research (Grant-in-Aid for JSPS Fellows).

REFERENCES

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

[2] David Corne, Marco Dorigo, and Fred Glover (1997) New ideas in optimisation. McGraw-Hill Publishing Company.

[3] Y. Ishida, H. Hirayama, H. Fujita, A. Ishiguro, K. Mori. (1998) Immunity-Based Systems and Its Applications. CORONA.

[4] D.E. Goldberg (1989) Genetic algorithm, search optimization and machine learning. Addison Wesley.

[5] 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.

[6] M. Yamamura, T. Ono, S. Kobayashi (1992) Character-Preserving genetic algorithms for Travering salesman problem. Jurnal of JSAI, Vol.7 No.6, pp.1049-1059, 1992.