New Ant System Algorithm by Ant-Tabu Agents

Taichi Kaji

Department of Information and Management Science,
Otaru University of Commerce, 3-5-21, Midori, Otaru, 047-8501, Japan
e-mail: kaji@res.otaru-uc.ac.jp

Abstract

The idea of ant system algorithm (AS) proposed by Dorigo is very unique. However, the standard type of the AS cannot obtain better solutions for random graphs. So, we design new agent by using intensification and diversification strategy, such as the tabu search is applied, in order to reach better solutions. We attempt to apply approach based on neighbourhood to the AS in terms of improving quality of solutions because the AS does not depend on neighbourhood. Furthermore, parallel ant system algorithm by above-mentioned new agents is implemented to reduce computational time. Finally we discuss the characteristics of new AS.

Keyword : Ant System Algorithm, Traveling Salesman Problem, Multi Agents, Parallel Algorithm, Meta-Heuristics, Tabu Search

1. Introduction

The ant system algorithm (AS) presented by Dorigo and others (Dorigo, Maniezzo and Colorni, 1996) is a new computational paradigm, which is a stochastic combinatorial algorithm. It can be able to solve the problem by "ants", that is, agents with very simple basic capability, which mimic the behavior of real ants. The ants could manage to establish shortest paths through the medium that is called "pheromone", used to communicate information regarding paths among individuals. The main characteristics of this paradigm are control by the combination of positive feedback through pheromone, and greedy heuristic. The greedy heuristic help find local good solution in the early stages of the search process. The pheromone promotes global search to escape local solutions by the communication among individual information regarding path.

This research is inspired by the problems studied by the ethologists (Deneubourg, Pasteels and Verhaeghe, 1983; Deneubourg and Goss, 1989; Nakamura and Kurumatani, 1995). For, example, it was to understand how ants could manage to establish shortest route paths from their colony to feeding sources and back. It is considered that real ants find shortest path by the communication process through the pheromone, which is very simple but exhibit highly structured behaviors and perform the complex tasks (Kawamura, Yamamoto, Mitamura, Suzuki and Ohuchi, 1998).

Dorigo and others applied his methodology to the Traveling Salesman Problem (TSP) and reported simulation results. In this paper, he compared the AS with the other meta-heuristics. He implemented Simulated Annealing (SA) and Tabu Search (TS), which are known as meta-heuristics and assessed using Oliver30 Problem for the comparisons. The results of the comparisons on Oliver30 showed that the AS was as effective as Tabu Search and better than Simulated Annealing (Dorigo, Maniezzo and Colorni, 1996).

However, when applying this AS the randomly generated graphs, there is a tendency for the solutions obtained using the AS to be trapped in bad near optimum. Therefore, we design a new agent by intensification and diversification strategy, such as tabu search is applied, in order to reach better solutions. Intensification and diversification are important component of tabu search (Glover and Laguna, 1997). We would like to describe how a combination of ant agent and strategy of intensification and diversification could apply. Furthermore, because the AS does not depend on neighbourhood, we apply an approach based on neighbourhood to above-mentioned tabu type of agents to search strongly in local regions of neighbourhood. And, we also attempt to reduce the computational time using parallel ant system algorithm by dividing colonies. This type can be done by partitioning the problem itself into several independent colonies. These algorithms are very well suited for implementation on a MIMD parallel mechanism. Finally, we analyze the performance of the proposed approaches and compare standard AS with proposed ant tabu approach. Herewith, we show that this proposed ant tabu approach gives a better quality of solutions than standard methodology. And, this parallel implementation based on dividing the colony naturally leads us the conclusion that save the computational time.

The approach by new agents discussed in this paper, designed to "solve" (in the sense of approximating the optimum) traveling salesman problem, is mathematically unexciting, but has performed remarkably well, both from the point of view of the computational effort involved, and from that of the quality of the solutions obtained on a variety of test problem. And, the agent heuristics approach will become an interesting vehicle in the solving of combinatorial optimization problems.

2. Traveling Salesman Problem

The traveling salesman problem (TSP) is typical optimization problem that is known as *NP*-complete and well used as benchmark in order to make the comparison with other heuristic approaches. In this problem the salesman must visit n cities, returning to his starting point, and is required to minimize the total cost of his trip. In going from city i to city j, he incurs a cost d_{ij}. That is, One seeks a minimal-cost Hamilton circuit on a complete graph having an associated cost matrix D. Entry d_{ij} in D is the cost of using the edge from the ith vertex to the jth vertex. The TSP arises in many different guises in operations research. One example is planning the movement of an automated drill press making holes at specified locations on printed circuit boards.

3. Ant System Algorithm

The paradigm of ant system algorithm (AS) proposed by Dorigo (Dorigo, Maniezzo and Colorni, 1996) is inspired by the problem to understand how almost blind animals like ants could manage to establish shortest route paths from their colony to feeding sources and back (Deneubourg, Pasteels and Verhaeghe, 1983; Deneubourg and Goss, 1989; Nakamura and Kurumatani, 1995). A moving ant lays some pheromone on the ground. The pheromone is used to communicate information among individuals regarding paths, and used to decide where to go. While an isolate ant moves essentially at random, an ant encountering a previously laid pheromone can detect it and decide with high probability to follow it, thus reinforcing the pheromone on the path with the ant's one. That is, the more ants follow the pheromone, the more attractive to be followed, since the probability with which an ant chooses a path increases with the number of ants that previously choose the same path. And, the quantity of pheromone on the shorter path grows faster than on the longer one. The final result is that very quickly all ant will choose the shorter path. From this phenomenon, idea of the AS is induced (Dorigo, Maniezzo and Colorni, 1996).

The AS has artificial ants on the computer for the TSP which has n towns and the length d_{ij} of path between towns i and j. Let m be the total number of ants over the all towns at time t. Each ant is a simple agent with the following characteristics:

- It chooses the town to go with a probability that is a function of the distance and of the amount of the pheromone information (The ant agent is not blind).
- It can not choose already visited towns until a tour is completed (The ant agent has some memory).
- It attaches the pheromone to the selected path.

Let τ_{ij} (t) be intensity of the pheromone information on edge (i, j) at time t. Each ant at time t chooses the next town, where it will be at time $t+1$. It needs n times to construct a completed tour, and an n time is one cycle (step) of the algorithm. The pheromone information is updated according to the following formula when a tour is completed (i.e., at one cycle).

$$\tau_{ij}(t+n)=\rho\ \tau_{ij}(t)+\Delta\tau_{ij}, \tag{1}$$

where ρ is a parameter such that $(1-\rho)$ represents the evaporation of pheromone between t and $t+n$, furthermore $\Delta\tau_{ij}$ is defined as follows.

$$\Delta\tau_{ij}=\sum_{\kappa=1}^{m}\Delta\tau_{ij}^{\kappa} \tag{2}$$

$$\Delta\tau_{ij}^{\kappa}=\begin{cases}\dfrac{Q}{L_{\kappa}} & \textit{if } \kappa\textit{-th ant uses edge } (i,\, j) \textit{ in its tour} \\ 0 & \textit{otherwize}\end{cases} \tag{3}$$

where Q is a given constant value and L_{κ} is the tour length obtained by the κ-th ant, and $\Delta\tau_{ij}^{\kappa}$ expresses the pheromone information laid on edge (i, j) by the κ-th ant between time t and $t+n$. In other word, the ant lays higher level pheromone on used edges if the tour length of the ant is shorter. The each κ-th ant has the tabu list $tabu^{\kappa}$ that saves the towns visited by κ-th ant, in order to forbidden to choose the towns already visited between time t and $t+n$. When a tour is completed, the $tabu^{\kappa}$ is then emptied and the ant is free again to choose. The transition probability from town i and town j for the κ-th ant is defined as follows.

$$p_{ij}^{\kappa}(\mathrm{t})=\begin{cases}\dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\cdot\left[\eta_{ij}\right]^{\beta}}{\sum_{\kappa\in\{V-tabu^{\kappa}\}}\left[\tau_{i\kappa}(t)\right]^{\alpha}\cdot\left[\eta_{i\kappa}\right]^{\beta}} & \textit{if } j\in\{V-tabu^{\kappa}\} \\ 0 & \textit{otherwize}\end{cases} \tag{4}$$

where τ_{ij} is the pheromone information that guides into global search, η_{ij} is defined as $1/d_{ij}$ that guides into greedy heuristic search, and α and β are parameters that control the relative importance of τ_{ij} and η_{ij}. Given the above-mentioned definitions, the Ant System Algorithm is showed as Figure 1. This standard ant system algorithm proposed by Dorigo is called Ant-Standard.

Ant System Algorithm :

$t=0$; $\tau_{ij}(t)=$ small positive value c ;
m ants are randomly positioned on the towns (starting points) ;
while (Stop Criterion $\neq$ True) do begin
    while (The tours are not completed) do begin
        for $\kappa=1$ to m do
            The κ-th ant choose the town j with probability $p_{ij}^{\kappa}(t)$,
            and move to the town j form town i ;
    end ;
    for $\kappa=1$ to m do
        Compute L_{κ} and $\Delta\tau_{ij}^{\kappa}$;
    Update Pheromone information for every edge (i, j),
        using equation (1) ;
    $t=t+n$;
end.

Figure 1. Ant System Algorithm

4. The Performance of Standard Ant System Algorithm

Dorigo and others have compared this standard ant system algorithm (Ant-Stndard) with other meta-heuristic algorithms. They implemented a simulated annealing (SA) and a tabu search (TS), and assessed their performances using the Oliver30 data. The results showed that the Ant-Standard for this problem was as effective as tabu search and better than simulated annealing in the Table 1 (Dorigo, Maniezzo and Colorni, 1996).

Table 1. Comparison of AS, TS, and SA by Dorigo

	Best	Average	Std.dev
AS	420	420.4	1.3
TS	420	420.6	1.5
SA	420	459.8	25.1

However, when applying this Ant-Standard algorithm the randomly generated graphs where the nodes are randomly laid in the unit square, there is a tendency for the solutions obtained using the Ant-Standard algorithm to be trapped in bad near optimum. The Table 2 and Figure 2 show the result for the randomly generated graphs in the range from 100 to 400

Table 2. Performance of Ant-Standard compared to SA on the randomly generated graph

Problem Size		100	200	300	400
Ant-Standard	Average	815.2	1158.8	1461.2	1652.6
	Best	800	1149	1456	1643
	Worst	829	1165	1478	1663
	Std. dev	10.30	5.74	16.77	8.13
SA	Average	786	1113	1381.2	1565.2
	Best	774	1099	1376	1546
	Worst	798	1127	1389	1577
	Std. dev	10.09	9.12	5.26	11.26

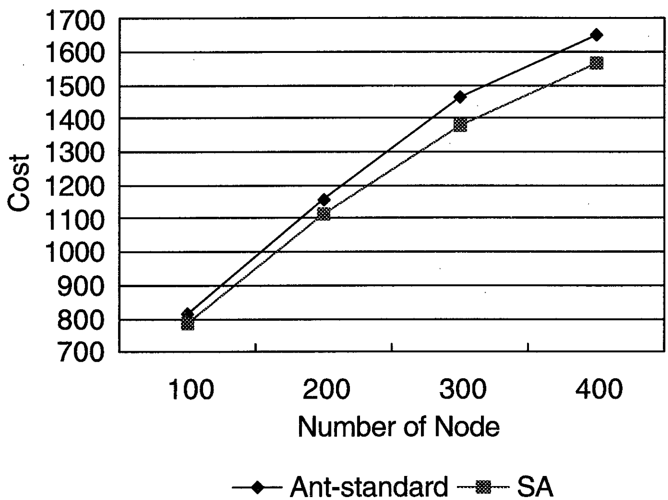

Figure 2. Comparison of Ant-Standard and SA

nodes, in comparison with simulated annealing. The AS algorithm is analyzed using parameter $Q=100$ and the number of ant agents is equal to the number of towns, as discussed by Dorigo and others (Dorigo, Maniezzo and Colorni, 1996). And, the simulated annealing used 2-opt neighbourhood structure. We have to consider the parameter's values of the simulated annealing in order to obtain good solutions in a reasonable amount of computational time (Ararts and Korst, 1989). The initial *tmp* (T) is 500 and the stop *tmp* (T) is 0.01, since, in spite of extending the above range of *tmp*, we did not obtain a significant improvement in the quality of the solutions. We found that *phi* = 0.92, which is cooling schedule parameter, gives an acceptable balance between solution quality and computational speed. It appears that initial length of the Markov chains $r=100$ and *tau* = 1.1, which control the length of Markov chains, yield good compromises between the quality of the solutions obtained and the time required. The data are average cost, best cost, worst cost, and standard deviation obtained after 10 repetitions of the both algorithms. The algorithms used here were coded in C++ and implemented on the Intel Celeron 400MHz. We see that standard ant system algorithm do not produce good results.

Next, the Figure 3 presents traces of run of the Ant-Stndard algorithm,

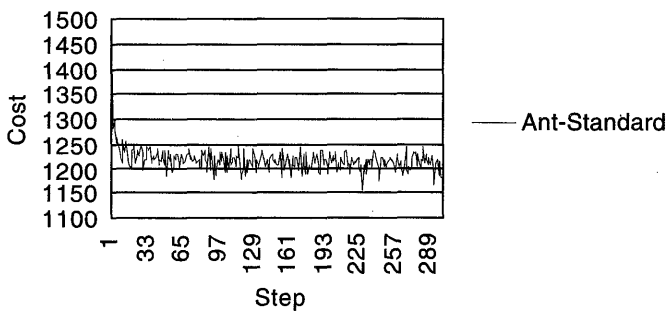

Figure 3. Trace of Ant-Standard

which is the length of the best-found tour at each cycle. The result show that the tour cost suddenly drops to low values in early steps, however, after that, the behavior is like the random walk and is never converged to optimal solutions.

5. Ant Tabu System Algorithm

Idea of the ant system algorithm (AS) is very interesting. However, the standard type of the AS cannot obtain better solutions for random graphs. So, we design new agent by using intensification and diversification strategy, such as tabu search (Glover and Laguna, 1997) is applied, in order to reach better solutions. Intensifications and diversifications are important components of tabu search. Intensification strategies are based on modifying choice rules to encourage move combinations and solution features historically found good. On the other hand, the diversification stage encourages the search process to examine unvisited regions and to generate solutions that differ in various significant ways from those seen before. We would like to describe how a combination of ant agent and strategy of intensification and diversification could apply.

First, we have to consider new agent with intensification strategy. Generally, the ants in this AS mutually have the communication using the pheromone to find better tour. The information of the tour obtained by each agent is communicated to the next population of agents through the medium of the pheromone and the next agents could search the better tour according to this medium. In the Ant-Standard algorithm, the pheromone information is defined as Q/L_K, which gives a higher value if the tour length is shorter. Figure 3 show that the behavior of tour length cost based on the communication by this way cannot converge, since the information of the best tour in each step is not strongly communicated through the pheromone.

So, we try to reinforce the pheromone information of best tour in each step to find better tour, as intensification strategy. That is, the pheromone information laid by elite agent that obtain best tour in each step is reinforced as $\sigma \cdot Q/L_{\kappa}$ to strongly communicate the information of better tour obtained by elite agent into next generation. This intensification strategy is based on modifying pheromone information of elite agent to encourage attractive regions to search better tours. Figure 4 shows traces of run of best-found solution in each step by the Ant-Standard and the AS with intensification strategy based on reinforcement of pheromone information. The results show that the performance of Ant-Standard gives fluctuation, while the AS using the reinforcement of pheromone information strongly converges to near optimal solution in the experiment.

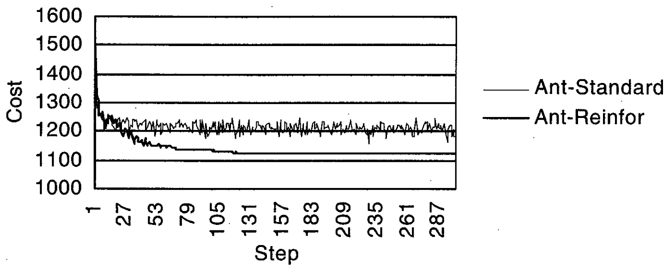

Figure 4. Trace of Ant-Reinfor compared to Ant-Standard
Ant-Standard: Normal AS
Ant-Reinfor: AS with Only Intensification Strategy

The intensification strategy in the AS shows remarkable improvement in contrast with the Ant-Standard. It seems that the reinforcement of the pheromone is important medium operation of communication. However, using only intensification strategy, the behavior fell to the situation in which best found solution in the later steps is similar to the best tour obtained before and it cannot improve in terms of quality of solution, because of the result of convergence strategy using the reinforcement of the pheromone information. This indicates that the system has ceased exploring new possibilities and no better tour will arise. The behavior of the restricted convergence for intensification strategy is induced by the fact that all agents have only identical ability, which is expressed by (4) equation. That is, the behavior of the AS with reinforcement of the pheromone information show strong convergence in the initial stages by means of the construction of the similar agents, however, the system lose the ability of the improvement later on and fall to stationary situation.

Therefore, we shall attempt to escape stationary situation by merging agents with different ability to the original population to give the system diversification, which means diversification strategy. The new agent for diversification strategy has longer-term memory, which is the strategy used in tabu search approach. The longer-term memory (Glover and Laguna, 1997) in tabu search is used to diversify the search compelling regions that are not visited before. Hence, we apply this idea to the ant agents. When either edge (i,j) or (j,i) is used by ant agent, we increase longer-term memory element $LMEM[i][j]\,;\, j>i$ *by* 1. Let $L_{ij}=C/LMEM[i][j]$, where C is a parameter. We call L_{ij} negative pheromone, which is signal not to attract agents. The ability of the new agent is formulated as

$$p_{ij}^{\kappa}(\mathrm{t})=\begin{cases}\dfrac{[\tau_{ij}(\mathrm{t})]^{\alpha}\cdot[\eta_{ij}]^{\beta}\cdot[L_{ij}]^{\gamma}}{\sum_{\kappa\in\{V-tabu^{\kappa}\}}[\tau_{i\kappa}(\mathrm{t})]^{\alpha}\cdot[\eta_{i\kappa}]^{\beta}\cdot[L_{i\kappa}]^{\gamma}} & if\ j\in\{V-tabu^{\kappa}\}\\ 0 & othersize\end{cases}\tag{5}$$

Then, C is set to 10, which gives better solution in experiment we tried. In

Figure 5, we show how the best-found solution in each steps decrease for the AS with only intensification strategy and the AS with both intensification strategy and diversification strategy. The results show that the AS with both intensification strategy and diversification strategy gives a better solution than the AS with only intensification strategy, by diversification for merging agents with different ability of longer-term memory. We see that diversification strategy is helpful to obtain better solutions.

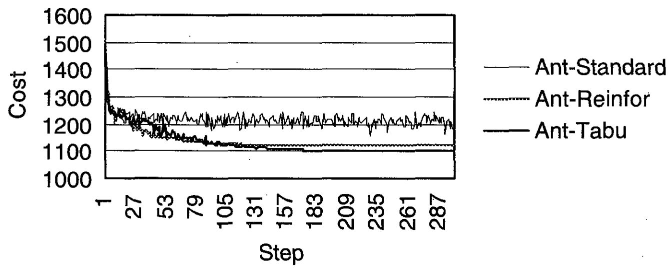

Figure 5. Trace of Ant-Tabu compared to other AS
Ant-Standard: Normal AS
Ant-Reinfor: AS with Only Intensification Strategy
Ant-Tabu: AS with Both Intensification Strategy and Diversification Strategy

Thus, the AS algorithm by new agents with abovementioned intensification strategy and diversification strategy is called ant tabu system algorithm (Ant-Tabu), this new agent is called ant-tabu agent. To assess the effectiveness of the proposed Ant-Tabu on randomly generated graphs, we evaluate the performance of the Ant-Tabu. The new agents with longer-term occupy 20% of colony. Table 3 shows an average cost, best cost, worst cost and standard deviation obtained after 10 repetitions of the Ant-Standard, the Ant-Tabu, and the simulated annealing (SA). We see that the Ant-Tabu outperforms the Ant-Standard in terms of solution quality. From this, we

Table 3. Performance of Ant-Tabu compared to Ant-Standard and SA

Problem Size		100	200	300	400
Ant-Standard	Average	815.2	1158.8	1461.2	1652.6
	Best	800	1149	1456	1643
	Worst	829	1165	1478	1663
	Std. dev	10.30	5.74	16.77	8.13
Ant-Tabu	Average	781	1099.8	1369.4	1560.4
	Best	778	1094	1356	1533
	Worst	786	1105	1380	1577
	Std. dev	2.82	3.54	9.30	15.6
SA	Average	786	1113	1381.2	1565.2
	Best	774	1099	1376	1546
	Worst	798	1127	1389	1577
	Std. dev	10.09	9.12	5.26	11.26

can get that combination of ant agent and strategy of intensification and diversification used in the tabu search is effective to obtain better solutions.

6. The Agents with Ability of Local Search

The solutions obtained using the ant system algorithm (AS) are derived by randomly selecting towns. Namely, this is a kind of Monte Carlo simulation technique. Necessarily, these solutions are not local solution that is a best solution in a feasible neighbourhood. More specifically, in the case of minimization, any solution i_{ant} obtained by the AS is not generally satisfy the condition for local solution such that $f(i_{ant}) \leq f(j)$, *for all* $j \in S_{i_{ant}}$, where $S_{i_{ant}}$ is neighbourhood (ex. 2-opt neighbourhood) for i_{ant}. Because AS does not depend on neighbourhood, we estimate it is helpful to apply algorithm based on neighbourhood to ant agent in order to obtain better solutions.

Therefore, there is a possibility of improvement in terms of quality of solution by giving the ability of local search to the ant-tabu agents. So, we attempt to solve the problem using ant-tabu agents with the ability of local search. That is, this ant is new ant-tabu agent with new characteristics which can search best solution in neighbourhood of the tour founded in colony. However, only agents that find better tours in colony carry out the local search characteristics. This revised ant-tabu system algorithm is called Ant-TSL. Thus, new ant tabu system algorithm has ability of search in both global and local region.

The tests were carried out again on the randomly generated graphs under above-mentioned computational environment. Although the number of agents for Ant-Tabu is equal to the number of towns, Ant-TSL uses only 100 agents. The results in Table 4 show that Ant-TSL, using ant-tabu agents with the local search system, gives a better solution than other AS and the SA. And the cost difference between best cost and worst cost is

Table 4. Performance of Ant-TSL compared to Ant-Tabu and SA

Problem Size		100	200	300	400
Ant-Standard	Average	815.2	1158.8	1461.2	1652.6
	Best	800	1149	1456	1643
	Worst	829	1165	1478	1663
	Std. dev	10.30	5.74	16.77	8.13
Ant-Tabu	Average	781	1099.8	1369.4	1560.4
	Best	778	1094	1356	1533
	Worst	786	1105	1380	1577
	Std. dev	2.82	3.54	9.30	15.6
Ant-TSL	Average	774	1091.5	1338.4	1515.2
	Best	774	1088	1336	1512
	Worst	774	1095	1342	1517
	Std. dev	0.0	3.5	2.24	1.72
SA	Average	786	1113	1381.2	1565.2
	Best	774	1099	1376	1546
	Worst	798	1127	1389	1577
	Std. dev	10.09	9.12	5.26	11.26

close. It is remarkable that the Ant-TSL, in spite of a little number of agents, outperforms standard Ant System Algorithm and the SA in terms of quality of solution. Our results indicate that the Ant-TSL, using new ant-tabu agents with ability of local search, can be used to obtain better near optimal for a wide variety of the TSP problem.

7. Parallel Ant System Algorithm

We also must evaluate the performance for CPU time of the proposed algorithm. The ant system algorithm (AS) is based on the mass of agents in a colony. Accordingly, the AS algorithm is time consuming because of processes of mass of agents. It is like the genetic algorithm, which is also time consuming because of computing for numbers of genes. This is serious bad point in the AS algorithm. Therefore, we attempt to reduce the computational time using parallel ant system algorithm by dividing colonies. This type can be done by partitioning the problem itself into several independent colonies. The algorithm is very well suited for implementation on a MIMD parallel mechanism. The parallel algorithm has been implemented on a network by MPI library (Gropp, Lusk and Skjellum, 1999). And, the computation by agent can be parallelized. This will generally require the partition of a set of the colonies into P processors. Each of these divided colonies is assigned to one process, which computes m/P tours. Next, the pheromone information of sub colonies is updated. That is, we divide the set of ant agents $\mathscr{C}$ into $\mathscr{C}_1, \mathscr{C}_2, \cdots, \mathscr{C}_p$ such that $|\mathscr{C}_i| = m/P$ for all i, $\mathscr{C}_1 \cup \mathscr{C}_2 \cup \cdots \cup \mathscr{C}_p = \mathscr{C}$, and $\mathscr{C}_i \cap \mathscr{C}_j = \phi$, for $i \neq j$, where P is the number of processor. Each partitioned colonies is assigned to each processor, which computes tours by ant agents in the $\mathscr{C}_i$, as Figure 6. After computing m/P tours, the pheromone information of each colony is determined and mutually

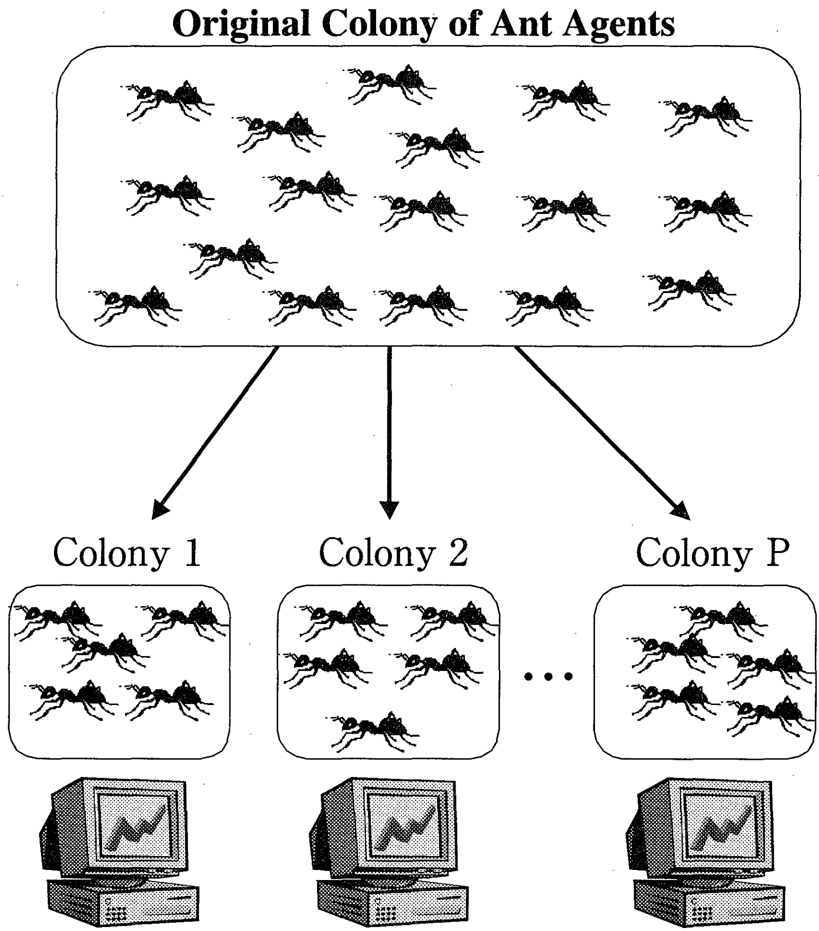

Figure 6. Parallel Ant System Algorithm

communicated among all processors together. The synchronization is required at each iteration step. Normally, this technique requires extensive communication since the synchronization is required at each cycle. However, the communication of parallel AS algorithm can be constructed using only pheromone information, without other information (i.e., a number of tours of agents in a sub colony). It is worth applying to only the problem in which the computation in one step is complex and time-consuming. So, we estimate that this proposed parallel algorithm allows us to reduce considerably the computational time.

Next, we evaluate the performance of the parallel ant system algorithm using 100 agents used in the Ant-TSL, with the number of nodes ranging from 100 to 400. Figure 7 illustrates the trace of CPU time required for sequential AS (cpu 1 line), two processors (cpu 2 line) and four processors (cpu 4 line) for the Parallel AS. The computational time required for the SA is about 600sec and do not remarkably increase. Although, that required for the AS algorithm is 110sec in the case of 100 nodes, the rate of an increase for computational time is high (as see cpu1 line in Figure 7) because of computation of mass of agents in a colony. However, In Figure 7 we see that the parallel AS can reduces computational time lineally as the number of processors increase. Running time is nearly 1/2 and 1/4 of CPU time of sequential AS for number of processor 2 and 4. There are various types of the AS by several kinds of strategy and parameter, and the running time is

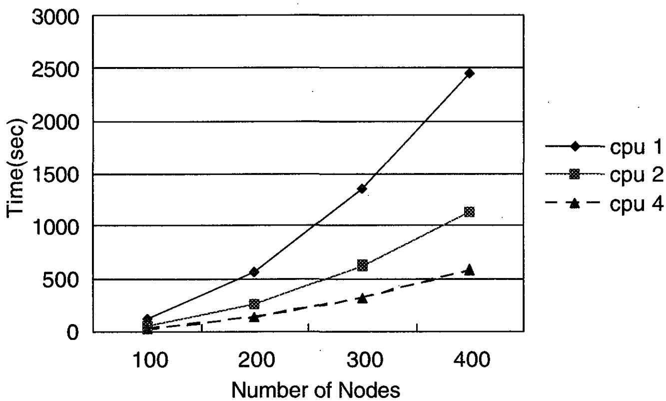

Figure 7. CPU Time for Parallel Ant System Algorithm

distinct. However, Other AS can also reduces computational time lineally as the number of processors. It should be noted that this parallelism for the AS allows us to reduce considerably the CPU time. It is possible that the AS obtain better solutions in reasonable computational time by parallel colony.

8. Conclusions

The ant system algorithm (AS) proposed by Dorigo is unique heuristics approach, which is the search activities over so-called ants, that is, agents with very simple basic capabilities which, to some extent, mimic the behavior of real ants. However, this standard type of the AS cannot obtain better solutions for TSP under random graphs. So, we proposed new agents with ability of intensification and diversification strategy, which are highly important components of tabu search. We illustrated combination of ant agents and strategy of intensification and diversification. We showed revised AS by combining these ant agents of tabu search type and local search to search strongly in local regions. It is helpful to apply idea based on neighbourhood to ant agents in terms of improving quality of solutions because the AS does not depend on neighbourhood. The computational results show that proposed algorithms outperform the standard ant system algorithm presented by Dorigo and the SA in terms of solution quality. We show that intensification and diversification strategy, and ability of local search in ant agent are helpful to reach better solutions.

Next, We also consider CPU time for AS which requires much more computational time. Therefore, we examined the parallel ant system algorithm by partitioning the colony itself into several independent colonies. This algorithm is well suited for parallel computation. It can communicate among each colony using only signals of pheromone information. Numerical results show that parallel ant system algorithm can reduces computational time lineally as the number of processors. It is possible that the AS obtain better solutions in reasonable computational time by parallel colony. In summary, our results indicate that the proposed ant tabu system algorithm, in conjunction with ideas of the reinforcement of pheromone as intensification strategy, the agent with longer term memory as diversification strategy and modified agents with ability of local search, can be used to obtain good-quality approximate solutions for a wide variety of TSP.

References

Ararts, E. and J. Korst. (1989). Simulated Annealing and Boltzmann Machines: John Wiley & Sons.

Deneubourg, J.L., J.M. Pasteels and J.C. Verhaeghe. (1983). "Probabilistic behavior in ants: a strategy of errors?." Journal of Theoretical Biology, 105, 259-271.

Deneubourg, J.L. and S. Goss. (1989). "Collective patterns and decision-making." Ethology, Ecology & Evolution, Vol.1, 295-311.

Dorigo, M., V. Maniezzo and A. Colorni. (1996). "The Ant System: Optimization by a colony of cooperating agents." IEEE Transactions on Systems, Man and Cybernetics-Part B, Vol.26, No.1, 1-26.

Glover, F. and M. Laguna. (1997). Tabu Search: Kluwer Academic Publishers.

Gropp, W., E. Lusk, and A. Skjellum. (1999). Using MPI: The MIT Press.

Kawamura, H., M. Yamamoto, T. Mitamura, K. Suzuki and A. Ohuchi. (1998). "Cooperative search based on pheromone communication for vehicle routing problems." IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E81-A, No.6, 1089-1096.

Nakamura, M. and K. Kurumatani. (1995). "A mathematical model for the foraging of an ant colony and pattern formation of the pheromone trial, Fundamental Theories of Deterministic and Stochastic Models in Mathematical Biology." The Institute of Statistical Mathematics Cooperative Research Report, 76, 120-131.