

Parameters Analysis for Basic Ant Colony Optimization Algorithm in TSP

Xianmin Wei

*School of Computer Engineering, Weifang University
5147 Eastern Dongfeng Street, Weifang 261061, China
wfxuweixm@126.com*

Abstract

In order to effectively address the lack of basic ant colony algorithm in terms of parameters, we use four-step method instead of the popular three-step, based on a large number of experiments of the parameters setting, this paper summed up an effective selection method for m , α , β , ρ and Q parameters to select the best combination of parameters. Applying the improved ant colony algorithms including optimal retention policy ant system, max-min ant system, ant-based sorting systems and best-worst ant system, performance comparison analysis was conducted with the same TSP problems, and experiments proved that the proposed method of parameter combinations greatly improves the speed of convergence.

Keywords: *Swarm intelligence, Ant colony algorithm, Parameter Selection, TSP, Visualization*

1. Introduction

Currently, the theoretical study field of swarm intelligence includes two main algorithms of ant colony algorithm (ACO) and particle swarm optimization algorithm (PSO). Ant colony algorithm, on behalf of swarm intelligence, has become a hot topic of today's distributed artificial intelligence research, which is first proposed in 1991 by the Italian scholar M. Dorigo, V. Maniezzo and A. Colorini [4], from the mechanisms of biological evolution inspired by the behavior of the real ants search path, which is also known as ant system (AS). M. Dorigo, who took full advantage of the similarity in the process between ant colony searching for food and famous traveling salesman problem (TSP), which absorbed the behavioral characteristics of ant colony, designed virtual ants to explore in different routes, and stayed virtual information contents [3] which gradually disappeared over time.

In computing process of ant colony algorithm, transmission of ant colony is guided by the strength of information amount left on to the paths. Path of ant colony movement is always close to the most informative path [1]. Researches of ant colony and ant colony algorithm show that paths of the strongest amount of information are closer to the optimal paths whether real ant colony or artificial ant colony system under normal circumstances. However, it still exists that the most informative path is not the optimal path required in the artificial ant colony system, this phenomenon often exists. Due to the artificial ant colony systems, the initial amount of information on the path is the same, the information obtained by ant colony on the first path is the distance information between the cities in TSP, for this time ant colony algorithm is equivalent to greedy algorithm. Information left for the first cycle of ants on the path does not necessarily reflect the most excellent direction of the path, especially when less number of individuals in the ant colony or more calculated path combination, there is no

reason to ensure that the first path ant colony created guided ant colony to the global optimal path. After the first iteration, information left by ant colony group because of a positive feedback made this path not optimal path, and may enhance path which may be very far from the optimal path, thereby to prevent ants from finding better global optimal path.

Not only for the first cycle the path established may be misleading the ant colony for any one cycle, the used information is more evenly distributed in all directions for this cycle, the release of pheromones may mislead decision-making later. Solutions identified by ant colony may be certainly enhanced by a method, pheromone released by ant colony as much as possible do not mislead the later ant colony [17].

At the same time, ant colony algorithm has shortcomings of long searching time and easy-to-stagnation. Recent studies show that there is a certain relationship between solution quality and the distance of the optimal solution. So the search was focused near the optimal solution found out in the search process, and it is basic emphasis for improving algorithm performance.

The ant colony algorithm (ACA) developed a simulation of ants foraging behavior of bionic optimization algorithm [6]. Algorithm through artificial ants randomly search the solution space, and the introduction of the pheromone evaporation and enhanced operation to guide the ant colony transfer, the volatilization mechanism of positive feedback, the algorithm has a certain optimization ability. The algorithm has a simple structure, which is easy to implement and nature of the characteristics of parallelism, Job-shop scheduling in recent years widely used, quadratic assignment, backpacks and other issues. Falling into a local extreme point easily for the basic ant colony algorithm, slow convergence insufficient in the literature on the basis of the ant colony system algorithm (ACS), the best - and worst ant swarm optimization (Best-Worst Ant System, BWAS) [5] and the largest - smallest ant colony algorithm (MMAS), and the above algorithms are analyzed and compared with.

The remainder of the paper is organized as follows: Section 2 describes the overview of basic ant colony algorithm, while Section 3 presents the detailed study of ant colony algorithm parameter settings. Then, Section 4 describes the experimental discussion for ant colony algorithm, and finally Section 5 concludes and shows the future work.

2. Ant Colony Algorithm Overview

The following details are advantages and disadvantages about ant colony algorithm.

2.1. Basic Principles of Ant Colony Algorithm

Ant colony algorithm can be expressed as follows, at the initial moment of the algorithm, m ants are randomly on the n cities, at the same time, and the first element of each ant's tab u list is set to the current city. Amount of pheromone on each path is equally to set $\tau_{ij}(0) = c$ (C is a small constant), according to the residual pheromone on path and heuristic information (between two cities distance), ants independently choose the next city at time t , the probability $p_{ij}^k(t)$ of ant k from city i to city j is:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{s \in J_k(t)} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Among them, $J_k(i) = \{1, 2, \dots, n\}$ - tab u_k denotes the city collection which ant k at next step is allowed to select. When the n cities joined list of tab u_k , ant k traversed once. The path ant k traversed is a feasible solution of TSP. In Eq.(1), η_{ij} is a heuristic factor, which denotes the desired degree for ants from city i to city j . In the AS algorithm, η_{ij} usually takes the

inverse of the distance between city i and city j. α and β respectively represents the relative importance of pheromone and heuristic factors [2]. When all ants completed a tour, information on each path is updated according to Eq.(2).

$$\tau_{ij}(t+n) = (1-\rho) * \tau_{ij}(t) + \Delta \tau_{ij} \quad (2)$$

$$\Delta \tau_{ij} = \sum_{k=1}^m \Delta \tau_{ij}^k \quad (3)$$

Where ρ ($0 < \rho < 1$) represents the evaporation coefficient of pheromone on the path, $1-\rho$ is the pheromone persistence coefficient. $\Delta \tau_{ij}$ said the pheromone increment of iterative edge (i, j). $\Delta \tau_{ij}^k$ in this iteration, the amount of pheromone left on the edge (i, j) by ant k. If ant k has not passed edge (i, j), then $\Delta \tau_{ij}^k$ is zero [7]. $\Delta \tau_{ij}^k$ can be expressed as follows:

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{if ant k passed the edge of ij in this tour} \\ 0, & \text{else} \end{cases} \quad (4)$$

Where Q is the normal number, L_k is the length of the path traversed by the ant k in the travel tour.

M. Dorigo put forwards three kinds of models of the AS algorithm [8], Eq.(4) is known as the ant-cycle, the other two models are called ant-quantity and the ant-density, main difference is in Eq.(4), namely in the ant-quantity model is as follows [9]:

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{d_{ij}}, & \text{ant k passed edge of ij at the moment of t and t+1} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

In the ant-density model is:

$$\Delta \tau_{ij}^k = \begin{cases} Q, & \text{ant k passed edge of ij at the moment of t and t+1} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

AS algorithm is actually the combination of a positive feedback principle and heuristic algorithm. When the path was chose, ants not only use the pheromone on the path but use the inverse of the distance between cities as a heuristic factor. The experimental results show that the ant-cycle model has better performance than the ant-quantity and ant-density models. This is because the ant-cycle model uses global information to update the amount of pheromone on the path, the ant-quantity and ant-density models use local information. AS algorithm's time complexity is $O(N_C * n^2 * m)$, space complexity is $S(n) = O(n^2) + O(n * m)$, of which N_C represents the number of iterations, n is the number of city, m is the number of ants [10].

2.2. Shortness of Ant Colony Algorithm

Many studies have shown that the AS algorithm has the strong ability to find better solutions, but there are the following defects:

(1) Local optimal solution, in terms of the nature of the solution on the algorithm, ant colony algorithm is looking for a better local optimal solution, rather than a mandatory global optimal solution [14].

(2) Stagnation problem in the middle of the work process (stagnation behavior), it is same as the fast convergence rate in the beginning of the algorithm, in the algorithm working process, after a certain number of iteration times, the ants may be able to produce stagnation near neighborhood in the local optimum solution [11].

(3) Longer search time, although compared with other algorithms, ant colony algorithm on the quality of the reconciliation and the number of iterations has certain advantages, but concerning the actual situation of the computer network, it will require a longer search time. Although computing speeds increasingly and parallel ant colony algorithm alleviate this problem to a certain extent, but for large and complex computer networks, this is a big hurdle [5].

3. Study of Ant colony Algorithm Parameter Setting

Ant colony algorithm achieved good results in TSP applications, but there are the following shortcomings:

At present, the parameter settings and properties research of basic ant colony algorithm are mostly still in the experimental stage, M. Dorigo [6] made a large number of experiments on ant system parameters and basic properties. Parameters discussed include:

m - number of ants.

α - pheromone relatively importance.

β - relative importance of heuristic factor.

ρ - pheromone evaporation coefficient ((1- ρ) indicating the pheromone persistence factor).

Q - amount of pheromone ants release.

In the experiment, in order to observe the impact of a parameter on the performance of the algorithm, when testing the parameters, other parameters take default values. The default value of each parameter are followings, $m = n * 1/4$ (in this paper it is 21), $\alpha = 1$, $\beta = 5$, $\rho = 0.5$, $Q = 100$ [13].

In the experiment, each data test is run for five times to take the average and optimal, TSP data used in the experiment from Eil51 urban problems, the number of iterations is 100 [16].

3.1. Hardware/software Platform

Hardware/software environment used in this experiment was, CPU 2.6GHz, 1GB memory, hard disk capacity of 280GB, Microsoft windows XP (Service Pack 3), development platform was the Microsoft Visual C++ 6.0 [15].

3.2. Effects of Ants Number on Basic ant Colony Algorithm

For effects of ant number m on the performance of the algorithm and its selection in practical application, this paper analyzed and determined through computer simulation, changes of ant number is {5,7,9,11,13,15,17,19,21,23,25,27,29,31}, effects and simulation results of ant number m on the performance of the algorithm were shown in Table 1 and Figure 1 below.

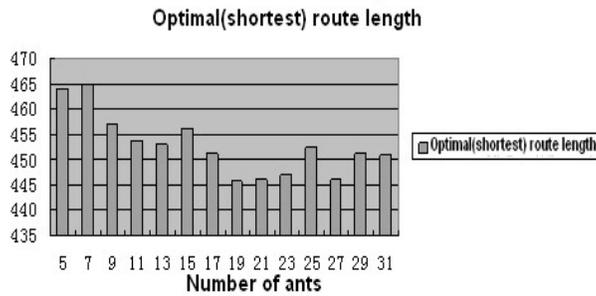


Figure 1. Simulation Results of Effects of Ant Number on Algorithm Performance

To choose the number m of ants in the ant colony algorithm, to comprehensively consider two indicators of algorithm global search ability and convergence speed, for issue-specific application conditions and requirements, to make reasonable or compromise selection in the global search ability and convergence speed. Figure1 is the experimental results, in which the horizontal axis indicates the number of ants, the vertical axis indicates the found optimal solutions. It can be seen from the figure when m is from $1/4$ to $2/5$ ant colony algorithm can find the optimal solution, through this experiment and other scholars' study, finally this paper drew out that the number of ant should be from $1/4$ to $2/5$, when the number of cities is smaller ant number m should be as close as possible to $2/5$ (if very small may consider $m=n$), when the number of cities is larger number of ants m should be as closely as possible to $1/4$, as considering two indicators of global search ability and convergence speed, this can find the optimal solution and better global search ability and convergence speed, performance of the algorithm is relatively stable.

3.3. Information Heuristic Factor and Expectations Heuristic factor

Information heuristic factor α reflects the relative importance of the amount of information accumulated in the ants' movement (*i.e.*, informative residues $\tau_{ij}(t)$), expectations heuristic factor β reflects the relative importance of heuristic information in the ants' movement (*i.e.*, the expectations η_{ij}) [11]. Expectations heuristic factor β reflects the effects strength of ant colony in priori of path search and uncertainty factors, greater the value, the greater the likelihood for ants to select local shortest path on a local point, although the search convergence rate speed up, however, ant colony in search of the optimal path weaken randomness to fall into local optimum easily; the size of the information heuristic factor α reflects the strength of stochastic factors in the path search of ant colony, greater the value, greater the likelihood of the re-choosing path, this weakened the randomness of search, when α value is too large, this can also make ant colony search prematurely trapped in local optima.

Global optimization of ant colony algorithm, at first the search process of ant colony must have a strong randomness; for fast convergence of ant colony algorithm, this also requires ant colony search process must have a high uncertainty. Both performance and role of ant colony algorithm is to complement each other and closely related. To get a good result it should choose the appropriate range of α and β , generally $\alpha = 0.5\sim 5$, $\beta = 1\sim 5$ [11]. Because the two parameters complement each other and closely related to, α and β combinations are used to discuss its impact on the performance of ant colony algorithm. α is $\{0.5, 1, 2, 5\}$, β is $\{1, 2, 5\}$, simulation and experimental results of the combination were shown in Table 2 and Fig.2.

Table 1. Effects Results of Ant Number on Algorithm Performance

Ant number	Optimal(shortest) route length	Running time(s)
5	463.987076	8.828
7	465.009199	14.11
9	457.001478	9.781
11	453.741327	13.187
13	452.822716	14.172
15	456.116011	13.531
17	451.117321	12.75
19	445.624177	14.187
21	446.078695	13.188
23	447.001731	13.921
25	452.299202	13.11
27	446.078695	21.468
29	451.223204	14.171
31	450.776659	13.515

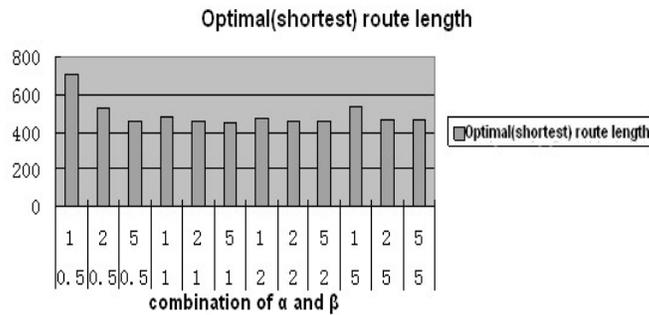


Figure 2. Simulation Results of α and β Combinations on the Algorithm Performance

The above experiment tells us that if to increase the fast convergence of the ant colony algorithm, and also to have a high certainty in ant colony search process, it is necessary to select both α and β , because their influence and role on the performance of ant colony algorithm is to complement each other and closely related to. On the basis of a large number of experiments in order to get better results, $\alpha = 1$, $\beta = 5$.

4. Experimental Discussion for Ant colony Algorithm

The followings are performance comparison of the most popular ant colony algorithms.

4.1. Improved Ant Colony Optimization Algorithm

In order to overcome the defect of AS, many scholars have studied and made a number of improvements, the following describes the five kinds of ant colony optimization algorithm.

Table 2. Algorithm Performance Results of α and β Combinations

Pheromone density factor α	Heuristic Information factor β	Optimal(shortest) route length	Running time(s)
0.5	1	703.903156	14.265
0.5	2	531.142891	19.203
0.5	5	458.382479	17.797
1	1	481.415788	17.718
1	2	461.263149	15.64
1	5	447.001731	12.093
2	1	475.705652	23.703
2	2	458.060510	12.062
2	5	459.590417	12.406
5	1	538.124487	12.703
5	2	465.841233	24.25
5	5	465.767435	24.375

(1) The reserved optimal solution ant system

Using the best ant can improve the quality of the solution in the ant system [10]. In the reserved optimal solution ant system (Elitist Ant System, EAS), after each iteration, the global optimal solution is further used, that is, as the pheromone is updated, just like many of the most excellent ants choosing this path. Compared with AS algorithm, AS-elite algorithm strengthens the use of an optimum solution when updating the pheromone, the pheromone update strategy is:

$$\tau_{ij}(t+1) = (1 - \rho) * \tau_{ij}(t) + \Delta \tau_{ij} + \tau_{ij}^*, \quad \rho \in (0,1) \quad (7)$$

$$\Delta \tau_{ij} = \sum_{k=1}^m \tau_{ij}^k \quad (8)$$

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L^k}, & \text{if the ant } k \text{ passed through edge of } ij \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

$$\Delta \tau_{ij}^* = \begin{cases} \sigma \cdot \frac{Q}{L^{sb}}, & \text{if edge of } (ij) \text{ is part of the optimal solution} \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

$\Delta \tau_{ij}^*$ is the increased amount of pheromone of the best ant on edge (i, j), σ is the optimal number of ants, L^{sb} is the global optimal solution.

(2) Ant colony system (ACS)

Ant colony system (ACS) [12] is one of the most successful follow-up algorithms, the main difference from the AS algorithm is: (1) Selecting the next city, ACS using the current better solution; (2) Adding the pheromone to the edge of local optimal solution; (3) Every time ants move from city i to city j, the pheromone on the edge (i, j) elements will appropriately reduce.

(3) Max-min ant system

Max-min Ant System (MMAS) [13] from another point of view of AS directly updates, which modifies the pheromone update model of AS, only to adjust the amount of information on the best individuals walking path in each generation, and speed up the convergence.

Concentration of pheromone on each path is limited in the range $[t_{\min}, t_{\max}]$, so that the amount of information can be effectively avoid much larger than or much less on a particular path than the rest of the path from happening, all of the ants concentrated on the same path, so that the algorithm is no longer diffusion, and speed up the convergence. In addition, the initial value of pheromone is set to the ceiling of the value, which helps increase the search capabilities of the algorithm at the initial stage, it is the best ant colony algorithm to solve TSP, QAO and other issues.

(4) Rank-based ant system (RAS)

Rank-based Ant System (RAS) proposed by Bernd Bullnheimer [14] was another expansion algorithm of AS. After each iteration in RAS, the paths ants passed will be sorted by order from small to large, that is $L^1(t) \leq L^2(t) \dots \leq L^m(t)$, according to the path length different weights were given. The shorter path length is, the greater the weight is. Global optimum weight was w , the weight of the r optimal solution was $\max\{0, w-r\}$, according to Eq.(11) to update pheromone on the path:

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \sum_{r=1}^{w-1} (w-r) \cdot \Delta \tau_{ij}^r(t) + w \cdot \Delta \tau_{ij}^{gb}(t), \quad \rho \in (0,1) \quad (11)$$

$$\Delta \tau_{ij}^r(t) = 1 / L^r(t), \Delta \tau_{ij}^{gb}(t) = 1 / L^{gb}$$

(5) Best-worst ant system(BWAS)

Best-worst ant system(BWAS) model tried to use the concept of evolutionary algorithms to improve the performance of the ACO model, the proposed BWAS uses AS transfer rules:

$$P_k(Y, S) = \begin{cases} \frac{[\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}{\sum_{\mu \in J_k(r)} [\tau_{r\mu}]^\alpha \cdot [\eta_{r\mu}]^\beta}, & \text{if } s \in J_k \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

τ_{rs} is the pheromone value of the boundary (r, s) , η_{rs} is inspire value, $J_k(r)$ is the node set left by ants K accessed, α, β are weights of value. The common volatile rules of AS is $\tau_{rs} \leftarrow (1-\rho) \cdot \tau_{rs}, \forall r, s, \rho \in [0, 1], \tau_{rs}$ is the pheromone evaporation parameter. In addition, BWAS considered the role of the following three new processes; the following is in-depth analysis [6].

4.2. Comparative Analysis of Simulation and Algorithm Performance

Compared with the AS algorithm, the common in above algorithms is to strengthen the use of the optimal solution. ACS algorithm and MMAS algorithm only enhanced the pheromone of optimal solution (global optimum, or iterative optimal) belonging to the path. RAS algorithm gives different weights according to the length of each iteration path, the shorter the path, greater weight.

It can be seen from the above data in this paper: when the problem size is relatively small, the results of various algorithms for solving are relatively close and relatively satisfactory. However, with the size of the problem gradually becoming larger, the gap between algorithm and known optimal results gradually increases. But, solving performance of BWAS and MMAS is better, basic AS algorithm without elite strategy is the worst, EAS with elite

strategy has greatly improved solution quality, but still has poor performance than MMAS ACS, BWAS and RAS, details shown in Tables 3 and 4.

Table 3. Compared Results for Solving Different TSP Instances Using AS, EAS, RAS

Problem		AS	EAS	RAS
Eil51	Optimal solution	426	426	426
	Number of iterations	1	1	1
	Iteration time	0	0	0
	Average solution	428.3	427.8	428.0
	Average iteration time(s)	0.003100	0.001600	0.001600
	Average total time(s)	0.018800	0.017100	0.018190
KroA100	Optimal solution	21282	21282	21282
	Number of iterations	1	1	1
	Iteration time	0	0.015000	0
	Average solution	21311.6	21317.0	21285.3
	Average iteration time(s)	0.012600	0.015600	0.012400
	Average total time(s)	0.018800	0.025000	0.024900
D198	Optimal solution	15858	15853	15829
	Number of iterations	1	1	1
	Iteration time	0.016000	0.031000	0.016000
	Average solution	15952.0	16093.0	16037.9
	Average iteration time(s)	0.018600	0.017300	0.017500
	Average total time(s)	0.025000	0.025000	0.022000

5. Conclusion

A large number of experiments illustrates that pheromone residual factor $1-\rho$, information heuristic factor α , expectation heuristic factor β , pheromone strength Q and ant number m are very important parameters, the way and principle of their selection will directly impact on global convergence and solution efficiency for ant colony algorithm, also the choice principle of these parameters was obtained by experiment and proposed an effective method to choose the optimal combination of parameters for ant colony algorithm as follows.

(1) Determine the number m of ants, based on the size selection strategies of the cities/ant $\approx 1/4$ to $2/5$ to determine the total number of ants.

(2) Parameter fine-tuning to adjust the smaller range of values of pheromone residual factor $1-\rho$ (0.1 as increments), in order to get good results, $1-\rho$ is 0.

(3) Parameters middle-tuning, that is, adjust the moderate range of values of heuristic factor α , and expect the heuristic factor β (1 as increments), and α, β use combinations to get the ideal solution, in order to get good results, $\alpha = 1, \beta = 5$.

(4) Parameters coarse-tuning, that is, adjust a larger range of values of the pheromone strength Q parameter (10 or greater number of adjustments) to get the preferred solution, in order to get good results, $Q = 100$.

The choice of α, β, m and ρ was based on experiment and other literature; in addition, the principle of six ant colony algorithm are analyzed and compared, and experimental results introduced performance of these ant colony algorithms for solving TSP problems, a comparative analysis on the performance of the algorithm was carried out, also obtained

optimal results of TSP problem when TSP problem optimal solution was the same, based on other properties (the number of iterations, the iteration time, etc.).

Under other conditions, that parameters setting study for various ACO algorithms using GA, PSO and Bee-inspired algorithms is our future work.

Table 4. Comparable Results for Solving Different TSP Instances Using MMAS, BWAS, ACS

Problem		MMAS	BWAS	ACS
Eil51	Optimal solution	426	426	426
	Number of iterations	1	1	1
	Iteration time	0	0	0
	Average solution	428. 0	427. 9	428. 5
	Average iteration time(s)	0. 006100	0. 003200	0. 001500
	Average total time(s)	0. 021700	0. 017200	0. 007900
KroA100	Optimal solution	21282	21282	21282
	Number of iterations	1	1	1
	Iteration time	0. 016000	0	0
	Average solution	21317. 0	21307. 6	21323. 7
	Average iteration time(s)	0. 014100	0. 007900	0. 014100
	Average total time(s)	0. 023500	0. 018800	0. 017200
D198	Optimal solution	15836	15847	15847
	Number of iterations	1	1	1
	Iteration time	0. 032000	0. 015000	0. 025000
	Average solution	15959. 0	16075. 8	16113. 2
	Average iteration time(s)	0. 019000	0. 018000	0. 015400
	Average total time(s)	0. 022000	0. 021800	0. 021800

Acknowledgements

This work is partly supported by Shandong Natural Science Foundation Program (ZR2011FL006), Shandong Science and Technology Development Program (2011YD01044), Shandong Spark Program(2012XH06005), and Weifang municipal Science and Technology Development Program (201301050).

References

- [1] T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning, IEEE Communications Surveys and Tutorials, vol. 10, no. 4, (2008).
- [2] H. Duan, D. Wang and X. YU, "Research Status and Future Prospects of the ant colony algorithm", China Engineering Science, vol. 9, no. 2, (2007).
- [3] L. Qin, "Improvement and Application of ant colony algorithm", Yangzhou University Master's Thesis, (2004).
- [4] M. Dorigo, V. Maniezzo and A Colorni, "Positive feedback as a search strategy", Politecnico di Milano, Italy, Technical Reportno, vol. 91, no. 16, (1991).
- [5] M. Dorigo, "Optimization Learning and Natural Algorithm", Ph.D. Thesis, Department of Electronics, Politecnico di Milano, IT, (1992).
- [6] A. Colorni, M. Dorigo and V. Maniezzo, "Distributed optimization by ant colonies", In: Proceedings of the 1st European Conference on Artificial Life, (1991).
- [7] H. Duan, D. Wang and J. Zhu, "Ant colony algorithm theory and applied research", Control and Decision, vol. 19, no. 12, (2004).

- [8] M. Dorigo, V. Maniezzo and A. Colomi, "The Ant System: Optimization by colony of cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics Part B, vol. 26, no. 1, (1996).
- [9] X. Hu, "Principle of ant colony optimization theory and its applications", Chongqing University Doctoral Thesis, (2004).
- [10] L. Machado and R. Schirru, "The Ant-Q Algorithm Applied to the Nuclear Reload Problem", Annals Nuclear Energy, vol. 29, no.12, (2002).
- [11] Q. Wu, J. Zhang and X. Xu, "Variability of ant colony algorithm", Computer Research and Development, vol. 36, no. 10, (1999).
- [12] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem", IEEE Transactions on Evolutionary Computation, vol. 1, no. 1, (1997).
- [13] T. Stotzle and H. Hoos, "The MAX-MIN ant system and local search for the traveling salesman problem", In: IEEE International Conference on Evolutionary Computation and Evolutionary Programming, Indianapolis, (1997) October 13-19, Seattle, USA.
- [14] B. Bullnheimer, R. F. Hartl and C. C Strauss, "A new rank-based version of the Ant System: A computational study", Central European Journal for Operations Research and Economics, vol. 7, no. 1, (1999).
- [15] O. Cordón, F. Herrera, I. Fern, Ez de Viana and L Moreno, "A New ACO Model Integrating Evolutionary Computation Concept: The Best-Worst Ant System", Lecture Notes in Computer Science,1838, (2000).
- [16] M. Florence, L. Manuel, S. Thomas and D. Yves, "Experimental analysis of pheromone-based heuristic column generation using I race", 8th International Workshop on Hybrid Metaheuristics, (2013) May 23-25, Ischia, Italy.
- [17] Jayadeva, S. Sameena, B. Amit, K. Ravi and C. Suresh, "Ants find the shortest path: A mathematical proof", Swarm Intelligence, vol. 7, no. 1, (2013).

Author



Xianmin Wei, he received the M. Sc. degree in computer applications from Shandong Science and Technology University (2005). He is currently an associate professor in school of computer engineering at Weifang University, China. He has published over 30 papers and 3 books in professional fields. Since 2011, he has been a member of IEEE-CS, ACM and CCCF, respectively. His fields of research are focused on swarm intelligent, intelligent sensor networks.

