International Journal of Hybrid Information Technology
Vol. 9, No.9 (2016), pp. 323-334
http://dx.doi.org/10.14257/ijhit.2016.9.9.30

Research on an Improved ACO Algorithm Based on Multi-Strategy for Solving TSP

Mengxing Li[1] and Zhuo Wan[2*]

[1]*School of Communication and Electronic Engineering, Hunan City University, Yiyang, Hunan 41300 China*
[2]*School of Communication and Electronic Engineering, Hunan City University Yiyang, Hunan 41300 China*

Abstract

Ant colony optimization (ACO) algorithm is a metaheuristic inspired by the behavior of real ants in their search for the shortest path to food sources. The ACO algorithm takes on these characteristics of robust, positive feedback distributed computing, easy fusing with other algorithms. But the basic ACO algorithm has some deficiencies of premature and stagnation phenomenon in the evolution process, and is easily trapped into local optimal solution. And it is difficult to explore other solutions in the neighbor space. So a improved ACO(DPSEMACO) algorithm based on dual population strategy, bi-directional dynamic adjust evaporation factor strategy of the pheromone and parallel strategy is proposed to solve the traveling salesman problem(TSP). In the DPSEMACO algorithm, the ants are divided into the two subpopulations by borrowing the mutual cooperation mechanism of biological community, which evolve separately and exchange information timely. The bi-directional dynamic adjusting evaporation factor strategy of the pheromone is used to change the corresponding path pheromone of different subpopulations in order to avoid to trap into a local optimum. The parallel strategy can avoid falling into a local optimum. And the DPSEMACO algorithm can expand the search space and improve the overall searching performance by repeated changing the pheromone of the each subpopulation and adaptive adjusting evaporation factor. Finally, in order to prove the optimization performance of the proposed DPSEMACO algorithm, some classic TSP instances are selected from the TSPLIB in this paper. And some existing methods are selected to compare the optimization performance with the proposed DPSEMACO algorithm. The experimental results demonstrate that the proposed DPSEMACO algorithm is feasible and effective in solving TSP, and takes on a good global searching ability and high convergence speed.

***Keywords**: Ant colony algorithm, dual population, evaporation factor, traveling salesman problem, optimization*

1. Introduction

Ant colony optimization(ACO) algorithm is a new evolution algorithm, which is inspired by the real ant colony behavior in nature. It is proposed by Dorigo at the beginning of 1990s[1]. It is applied to solve the complex combination optimization problem[2]. At the same time, it is also used in the job shop scheduling problem (JSP), quadratic assignment problem(QAP) and so on[3-5]. The ACO algorithm has achieved a series of good results in solving these problems. It takes on better robustness, parallel distributed computing and easy combining with other heuristic methods. In the short term, it has been greatly developed, and its application fields are also expanding. These show that the ACO algorithm has some advantages in solving complex combinatorial optimization problems[6,7].

ISSN: 1738-9968 IJHIT

However, the ACO algorithm exists long search time, easy falling into local optimum, and low computational time and so on. In order to avoid stagnation or premature phenomenon, many scholars proposed a lot of improved ACO algorithms. Abadeh *et al*.[8] proposed an evolutionary algorithm to induct fuzzy classification rules. The algorithm uses an ant colony optimization based local searcher to improve the quality of final fuzzy classification system. Borkar and Das[9] proposed the MAF-ACO algorithm, which emulates the foraging behavior of ants found in nature. The components of the MAF-ACO algorithm as stochastic approximation algorithms is viewed and the ordinary differential equation (o.d.e.) method is used to analyze their convergence. Zhang *et al*.[10] proposed a hybrid optimization algorithm with particle swarm optimization(PSO) and ant colony optimization (ACO). In the proposed algorithm, the PSO is used to optimise the parameters in the ACO, which means that the selection of parameters does not depend on artificial experiences or trial and error, but relies on the adaptive search of the particles in the PSO. Twomey *et al*.[11] proposed a study in which we analyze the impact that different communication policies have on the solution quality reached by a parallel homogeneous multi-colony ACO algorithm for the traveling salesman problem. Manuel and Christian[12] proposed a Beam-ACO algorithm, which is a hybrid method combining ant colony optimization with beam search. In general, Beam-ACO algorithms heavily rely on accurate and computationally inexpensive bounding information for differentiating between partial solutions. Meena *et al*.[13] proposed a parallel ACO algorithm o select features for categorising longer documents to closely related categories. Heuristic value for each word is computed by the statistical dependency of the term to a category and its compactness value. Zhang *et al*.[14] proposed a physical topology awared Chord model (Ant-Chord) based on ant colony algorithm. The ideas of Ant-Chord is to regard the storage nodes in the whole Chord as a TSP problem and solve the TSP problem quickly by using the ant colony algorithm. Shuang *et al*.[15] proposed a hybrid PS-ACO algorithm based on ACO algorithm modified by particle swarm optimization(PSO) algorithm. The pheromone updating rules of ACO are combined with the local and global search mechanisms of PSO. Medina-Rodriguez *et al*.[16] proposed an efficient solution to determine the best sequence of G commands of a set of holes for a printed circuit board in order to find the hole-cutting sequence that shortens the cutting tool travel path. Kötzing *et al*.[17] contributed to the theoretical analysis of ant colony optimization and studies this type of algorithm on one of the most prominent combinatorial optimization problems, namely the traveling salesperson problem (TSP). We present a new construction graph and show that it has a stronger local property than one commonly used for constructing solutions of the TSP. Janaki Meen *et al*.[18] formulated the text feature selection problem as a combinatorial problem and proposed an enhanced ACO algorithm to find the nearly optimal solution for the same. Ugur and Aydin[19] proposed an extra data structure that we called best tours graph feeding the pheromone trail information for ACO algorithms. Best tours graph is a table that blends the information on the global best tours encountered statistically during iterations and includes the strengths of edges. Guo and Liu[20] analyzed and compared several typical Ant Colony Optimization algorithms which employ different improving methods, and then conclude three most widespread sorts of strategies(improvement on the construction of solutions, the update of pheromone trails and the treatment of solutions) from them. Pintea *et al*.[21] proposed a new parallel computing technique based on ant colony optimization for a dynamic routing problem. The new technique uses a parallel model for a problem variant that allows a slight movement of nodes within their neighborhoods. Cheng and Xiao[22] proposed one kind of dynamic positive and negative feedback ACO which differs from existing ACO in two important aspects:

(i) positive feedback inner-colony and negative feedback inter-colony, and (ii) parallel implementation on Haloop, a framework built with iterative Map Reduce model. Positive and negative feedback coefficient will change the weights of exclusive pheromone and attractive pheromone to lead the transition from competition to cooperation occurring dynamically and gradually. Zhang *et al*.[23] proposed a multiobjective EA, *i.e.,* MOEA/D-ACO based on combining ant colony optimization(ACO) and the multiobjective evolutionary algorithm (EA) with decomposition(MOEA/D). MOEA/D-ACO decomposes a multiobjective optimization problem into a number of single-objective optimization problems. Skinderowicz *et al*.[24] proposed Population-based ant colony optimization (PACO), which is one of the most efficient ant colony optimization (ACO) algorithms. Its strength results from a pheromone memory model in which pheromone values are calculated based on a population of solutions. Elloumi *et al*.[25] proposed a new hybrid method(PSO-ACO) based on particle swarm optimization(PSO) and ant colony optimization (ACO) algorithms for solving the traveling salesman problem (TSP). The new hybrid method(PSO-ACO) is validated using the TSP benchmarks and the empirical results considering the completion time and the best length, illustrate that the proposed method is efficient. de Souza and Pozo [26] proposed a multiobjective evolutionary algorithm based on decomposition and ant colony optimization(MOEA/D-ACO) and a graphics processing unit (GPU) implementation of MOEA/D-ACO using NVIDIA CUDA (Compute Unified Device Architecture) in order to improve the execution time. Jiang[27] proposed a novel hybrid ant colony genetic (NHACG) algorithm with recent patents based on integrating multi-population strategy and collaborative strategy for the complementary of ant colony optimization(ACO) algorithm and genetic algorithm (GA). Some traveling salesman problems (TSP) are selected to test the effectiveness of the NHACG algorithm.

In this paper, the dual population strategy, bi-directional dynamic adjust evaporation factor strategy of the pheromone and parallel strategy are introduced into the basic ACO algorithm in order to avoid the premature and stagnation phenomenon, to fall into local optimal solution. And an improved ACO(DPSEMACO) algorithm based on combining the dual population strategy, bi-directional dynamic adjust evaporation factor strategy of the pheromone and parallel strategy is proposed. Some classic TSP instances are selected the TSPLIB standard library to prove the optimization performance of the proposed DPSEMACO algorithm.

The rest of this paper is organized as follows. Section 2 briefly introduces ant colony optimization(ACO) algorithm. Section 3 briefly introduces dual population strategy, bi-directional dynamic adjust evaporation factor strategy of the pheromone and parallel strategy for improving the basic ACO algorithm. Section 4 briefly introduces the flow describing of DPSEMACO algorithm. Section 5 gives experiment for TSP and results analysis. Finally, the conclusions are discussed in Section 6.

2. Ant Colony Optimization (ACO) Algorithm

The ACO algorithm is a metaheuristic inspired by the behavior of real ants in their search for the shortest path to food[1]. The ACO algorithm consists of a number of cycles (iterations) of solution construction. In each iteration, a number of ants construct complete solutions by using heuristic information and the collected experiences of previous groups of ants. These collected experiences are represented by the pheromone trail which is deposited on the constituent elements of a solution. Small quantities are deposited during the construction phase while larger amounts are deposited at the end of each iteration in proportion to solution quality. Pheromone can be deposited on the components and/or the connections used in a solution depending on the problem.

In the ACO algorithm, the ACO algorithm simulates the optimization of ant foraging behavior. The procedure of the ACO algorithm is described in Figure 1.

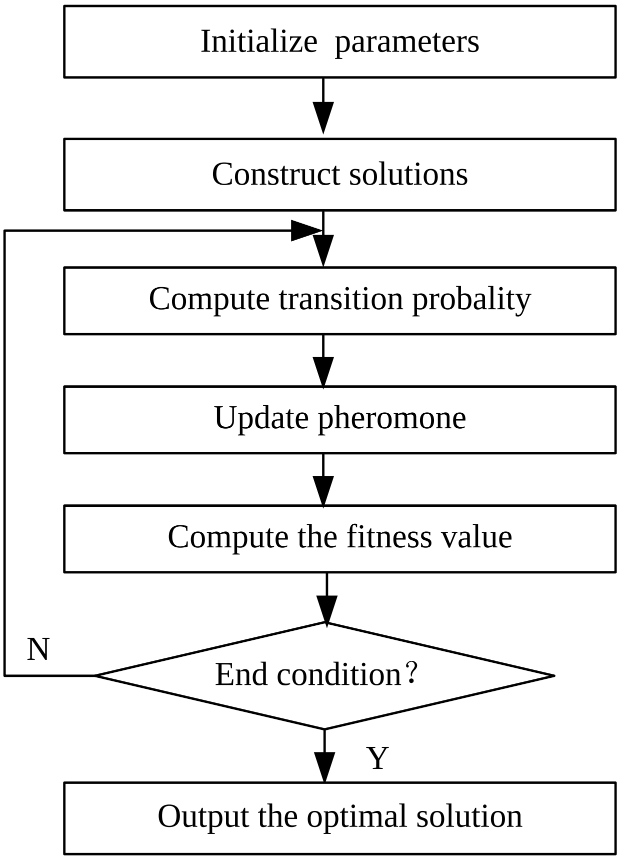

Figure 1. The Flow of the ACO Algorithm

The procedure of pheromone update rule is shown as follows:

(1) The transition rule

To visit the next city s with the probability $p_k(r,s)$,

$$p_k(r,s) = \begin{cases} \dfrac{\tau(r,s)^{\alpha} \cdot \eta(r,s)^{\beta}}{\sum_{u \in J_r^k} \tau(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}} & if \ s \in J_r^k \\ 0 & otherwise \end{cases} \quad if\, q > q_0 (BiasExploitation) \tag{1}$$

In two formula, $p_k(r,s)$ is the transition probability, $\tau(r,u)$ is the intensity of pheromone between city r and city u in the i^{th} population, $\eta(r,u)$ is the length of the path from city r to city u, J_r^k is the set of unvisited cities of the k^{th} ant in the i^{th} population, the parameter α and β are the control parameters, q is a uniform probability [0, 1].

(2) The pheromone update rule

In order to improve the solution, the pheromone trails must be updated. Trail updating includes local updating and global updating. The local trail updating formula is given by:

$$\tau(r,u) = (1-\rho)\tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \tag{2}$$

In here, ρ ($0 < \rho < 1$) is the pheromone trial evaporating rate. $\Delta\tau_k(r,s)$ is the amount of pheromone trail added to the edge(r,s) by ant *k* between time *t* and t+Δt in the tour. It is given by:

$$\Delta\tau_k(r,s) = \begin{cases} \frac{Q}{L_k} & (r,s)\in\pi_k \\ 0 & otherwise \end{cases} \tag{3}$$

where Q is a constant parameter, L_k is the distance of the sequence π_k toured by ant in Δt.

3. An Improved ACO Algorithm

The basic ACO algorithm is easy to fall into local optimal solution in the simulation results of solving TSP. After the search is executed a certain extent, the found the solution of all individuals are completely consistent, which will cause the stagnation phenomenon. Because this can not further search for solution space, it will not be able to find the global optimal solution. The main cause of this phenomenon is excessive accumulation pheromone of local path. When a lot of ants select the same path, the pheromone on this path will suddenly be increased to cause the larger difference of the pheromone between the path and the other path, so that subsequent ants would focus on this path. It will cause the blockage and stagnation. These conditions is shown that the ACO algorithm will occur the premature convergence and local convergence. Therefore, how to control the pheromone concentration is key method to find out a balance point between the convergence speed and the convergence space for the ACO algorithm. Some strategies of improvements for ACO algorithm are proposed in this paper.

3.1. Dual Population Strategy

In order to avoid the influence of the path selection of ants in the same population, that is more and more ants will choose the same path under the positive feedback, so that single population will be prone to the stagnation phenomenon in the late evolution. The dual population strategy is introduced into the basic ACO algorithm. The basic idea of the dual population strategy is described in here. The ants in the basic ACO algorithm is divided into two subpopulation in order to independently search and regularly exchange the excellent solution and information. Because the path selection of the ants in the same population is affected by the selection of other ants, more and more ants will choose the same path under the function of the same path. Thus, a single population is prone to the stagnation phenomenon in the later stage of evolution. The ACO algorithm based on the dual population can effectively restrain the stagnation phenomenon. Each subpopulation searches the solution path according to the respective probability and regularly exchanges the excellent solution and information in order to guarantee the diversity of the solutions in the ACO algorithm. The ACO algorithm based on the dual population can realize two solving basic problems: how to determine the conditions of information exchange, and the content and form of information exchange. In general, the information is exchanged by the interval of a number of iterations. The periodical exchange is more intuitive and easy to implement. But too frequent exchanges can be close to the single population, and too less exchanges can not reflect the superiority of dual population algorithm. So the exchange time may take m/5~m/10, and the exchange content is mainly reflected in the pheromone distribution. Because the pheromone distribution in the single population will tend to be consistent in the evolution, the most of pheromone are distributed on the small number of paths. The pheromone distribution is different in different population. So the pheromone in the excellent solution path can be dispersed in another population, so that the ants in the other population can break the stagnation state by the larger probability.

The ACO algorithm based on the dual population is shown in Figure 2. It can make the ants in the two subpopulations can break their stagnation state, greatly improve the search performance, and enhance the global optimization ability.

3.2. Bi-Directional Dynamic Adjustment Evaporation Factor Strategy

The size of pheromone evaporation factor ρ directly affect the pheromone distribution size of each path, which directly affect the global searching ability and convergence speed of the ACO algorithm. In the basic ACO algorithm, the ρ value is fixed value. If the ρ value is larger, the pheromone in these unvisited paths will reduce to be close to zero in dealing with the large scale problem, which is not conducive to find a better solution and easy to fall into local optimum. Based on the dual population strategy for improving the ACO algorithm, a bi-directional dynamic adjust evaporation factor strategy is proposed to further improve the ACO algorithm. For the pheromone evaporation factor ρ of two independent subpopulation, the value range of ρ is set between zero and one. In the initial stage of the algorithm, the smaller value of the ρ is set for one subpopulation($\rho = 0.1$), the larger value of the ρ is set for the other subpopulation($\rho = 0.9$). In the iterative process, when the solving is trapped into a local optimum, the ρ values of two subpopulations are dynamically adjusted. The adjusting method is that the smaller ρ value in the initial stage is gradually increased and the larger ρ value in the initial stage is gradually reduced. The different subpopulations can expand their search space and avoid excessive concentration on some search optimal paths by using the changes of ρ value in order to find a better path. For different subpopulation, due to the difference of the ρ value in the initial stage, the pheromone on the path of different subpopulation exists the larger difference in the iterative process. So when the algorithm falls into a local optimum, the pheromone exchange of the corresponding path of two subpopulations can take place the larger shock changes, so as to further expand the search space of the algorithm, and the search results converge to the global optimal solution.

Therefore, when the solving is trapped into a local optimum in the iterative process, the bi-directional dynamic adjustment evaporation factor strategy is used to adaptively adjust the ρ value. The evaporation factors of two subpopulations are ρ_1 and ρ_2, and the initial value of ρ_1 is smaller and the initial value of ρ_2 is larger.

$$\rho_1(t) = \begin{cases} \mu_1 \times \rho_1(t-1), \text{the solving is trapped into the local optimum in the t iteration} \\ \rho_1(t-1), otherwise \end{cases} \tag{4}$$

$$\rho_2(t) = \begin{cases} \mu_2 \times \rho_2(t-1), \text{the solving is trapped into the local optimum in the t iteration} \\ \rho_2(t-1), otherwise \end{cases} \tag{5}$$

where t is iterative time, μ_1 and μ_2 are dynamic adjustment evaporation factors for two subpopulations. According to the results of the experiment, $\mu_1 \in (1,1.5)$ and $\mu_2 \in (0.5,1)$ are better for the algorithm.

3.3. Parallel Strategy

According to the experimental results, the population is divided into two subpopulations(P_1 and P_2) according to the appropriate proportion. The ants in the subpopulation P_1 start from the same point to solve the shortest path between two points according to the objective function $F(t)$, and the ants in the subpopulation P_2 start from the same point to solve the shortest path between two points according to the objective function $F(t)$. The two subpopulations need to independently complete their tasks until one of two subpopulations finds the shortest path. In the process of solving the problem, this will guarantee even if the algorithm is trapped in a local optimum, it can also increase the diversity of the algorithm for meeting the requirements by using the subpopulation P_2. In order to ensure the relative independence between two subpopulations(P_1 and P_2), the multi thread parallel processing techniques are used in the multi core systems. At the same time, it also reflects the parallelism characteristics of the algorithm. In multi core system, there is a multi thread parallel mode, namely OpenMP mode, which uses the shared memory to save the memory space without installing additional system for ensuring the parallel processing. Therefore, this model is used to put the subpopulations into the parallel region, and allocate multiple threads for the ant colony. These threads are responsible for solving the shortest path among the subpopulations until the algorithm reaches the end condition, leaves the parallel region and output the global optimal solution at this time.

4. The Flow Describing of DPSEMACO Algorithm

The proposed DPSEMACO algorithm is used to effectively solve the shortest route problem, it is described as follow:

Step 1. Initialize

The parameters in the DPSEMACO algorithm need be initialized. These parameters are the maximum iteration times(G_{max}),the size of ants(m_1 and m_2),the pheromone factor(α) and heuristic factor(β), the initial pheromone evaporation factor(ρ_{01} and ρ_{02}), initial uniform probability (q_0),pheromone amount (Q_1 and Q_2), and so on.

Step 2. The m_1 ants in the subpopulation P_1 and m_2 ants in the subpopulation P_2 are randomly placed into the initial cities, and the selection probability for the next city is computed according to the formula in order to complete their traveling.

Step 3. The visited cities are added into Tabu list.

Step 4. The shortest path of two subpopulations(P_1 and P_2) are respectively recorded in this iteration.

Step 5. Enter the guaranteed optimal function. The obtained optimal paths by two subpopulations(P_1 and P_2) are compared in this iteration. And the shortest path is taken as the optimal path in this iteration.

Step 6. The Tabu list is modified. And the selected cities are added into the Tabu list.

Step 7. The pheromone on the optimal path is globally updated for two subpopulations(P_1 and P_2). The pheromone maximum value and pheromone minimum value are set at the same time.

Step 8. When the algorithm is trapped in local optimum, the communication function of the population is used to exchange the pheromone of the corresponding path of the subpopulations P_1 and subpopulations P_2.

Step 9. When the algorithm is trapped into local optimum, the values of the ρ_1 and ρ_2 are dynamically adjusted according to the equation(4) and equation(5).

Step 10. Set the iterative counter $t=t+1$. If $t<T_{max}$, return to Step 3. Otherwise, the DPSEMACO algorithm is terminated.

Step 11. Output the optimal path and the shortest distance.

5. Experiment and Analysis

In order to test the optimization ability of the proposed DPSEMACO algorithm for solving complex problem, 6 TSP datasets are selected form TSPLIB standard library in this paper. Under different parameter combination, the proposed DPSEMACO algorithm and basic ACO algorithm are compared for the 6 TSP datasets. The distance of each two cities is computed by using Euclidian distance. It is a very complicated problem to obtain the parameters' values of two algorithms, because the changes of parameters' values could seriously affect solving the optimum value. So the most reasonable initial parameters' values are obtained by testing and modifying. The obtained initial parameters' values are: ants $m_1 = m_2$ =30, pheromone factor $\alpha_1 = \alpha_2$ =1.0, heuristic factor β_1 =2.0 and β_2 =4.0, evaporation factor ρ_1 =0.10 and ρ_2 =0.90, dynamic adjustment evaporation factors μ_1 =1.1 and μ_2 =0.9. pheromone amount $Q_1 = Q_2$ =80, the maximum iteration times $G_{\max}$ =1000. The experiment environments are: the Pentium CPU 2.40GHz, 8.0GB RAM with Windows 7 and Matlab2010. Two algorithms are independently run 20 times. In this paper, two algorithms are independently run 20 times. The optimal value, average value and average number of iteration are selected to described the optimization performance of two algorithms. The numerical experiment results are shown in Table 1.

Table 1. The Numerical Experiment Results

TSP	Algorithm	Best Value	Optimal Value	Average value	Average iteration
att48	ACO	33522	33804	343561	482
	DPSEMACO		33723	**33524**	385
eil76	ACO	538	602	553	504
	DPSEMACO		557	**544**	386
ch130	ACO	6156	6204	6153	483
	DPSEMACO		6149	**6116**	352
rat195	ACO	2323	2429	2384	604
	DPSEMACO		2395	2349	496
rat783	ACO	8806	9398	9153	853
	DPSEMACO		9204	9042	692
d1291	ACO	50801	54482	52907	934
	DPSEMACO		53186	52379	817

As can be seen from Table 1, for 6 TSP datasets, the optimal value, average value and average number of iteration of the proposed DPSEMACO algorithm are better than these of the basic ACO algorithm. This shows that the DPSEMACO algorithm is very effective for solving the TSP problems. And TSP datasets of att48, eil76 and ch130 can find the best value that is close to the known optimal solution by using proposed DPSEMACO algorithm. This shows that the dual population strategy can greatly enhance the global search ability. And for solving larger scale instance, the numerical experiment results that the proposed DPSEMACO algorithm can obtain better optimization value. Therefore, the experiment results show that the proposed DPSEMACO algorithm is feasible and effective in solving TSP, and takes on a good global searching ability and high convergence speed.

6. Conclusion

The ACO algorithm is a bio-inspired optimization algorithm, which simulates the swarm intelligence behavior of ants. It takes on these characteristics of robust, positive feedback, distributed computing, easy fusing with other algorithms. But it exists premature and stagnation phenomenon and easy falling into local optimal solution, and it is also difficult to explore other solutions in the neighbor space. So the dual population strategy, bi-directional dynamic adjust evaporation factor strategy of the pheromone and parallel strategy are introduced into the basic ACO algorithm in order to evolve separately and exchange information timely, change the corresponding path pheromone of different subpopulations, expand the search space and improve the overall searching performance by repeated changing the pheromone of the each subpopulation and adaptive adjusting evaporation factor. Then an improved ACO(DPSEMACO) algorithm is proposed to solve the TSP in this paper. The goal is to prove the optimization performance of the proposed DPSEMACO algorithm in solving complex optimization problem. The optimal value, average value and average number of iteration of the proposed DPSEMACO algorithm are better than these of the basic ACO algorithm for 6 TSP datasets. The experiment results show that the proposed DPSEMACO algorithm is feasible and effective in solving TSP, and takes on a good global searching ability and high convergence speed.

Acknowledgments

This research was supported by Hunan Province Science and Technology Project Under Grant 2010FJ4128. The authors would like to thank all the reviewers for their constructive comments. The program for the initialization, training, and simulation of the proposed algorithm in this article was written with the tool-box of MATLAB 2010 produced by the Math-Works, Inc.

References

[1] M. Dorigo and M. Gambardellal, “Ant colony system: A cooperative learning approach to the traveling salesman problem”, IEEE Transactions on Evolutionary Computation, vol.1, no.1, **(1997)**, pp. 53-66.

[2] W. Deng, R. Chen, B. He, Y.Q. Liu, L.F. Yin and J.H. Guo, “A novel two-stage hybrid swarm intelligence optimization algorithm and application”, Soft Computing, vol.16, no.10, **(2012)**, pp.1707-1722.

[3] Y.G. Zhang, S.B. Zhang and Q.S. Xue, “Improved ant colony optimization algorithm for solving constraint satisfaction problem”, Journal on Communications, vol. 36, no.5, **(2015)**, pp.1-6.

[4] O. Baskan, S. Haldenbilen, H. Ceylan and H. Ceylan, “A new solution algorithm for improving performance of ant colony optimization”, Applied Mathematics and Computation, vol.221, no.1, **(2009)**,pp. 75-84.

[5] J.Q. Geng, L.P. Weng and S.H. Liu, “An improved ant colony optimization algorithm for nonlinear resource-leveling problems”, Computers and Mathematics with Applications, vol.61, no.8, **(2011)**, pp. 2300-2305.

[6] L.N. Xing, P. Rohlfshagen, Y.W. Chen and X. Yao, “A hybrid ant colony optimization algorithm for the extended capacitated arc routing problem”, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 41, no.4, **(2011)**, pp. 1110-1123.

[7] W. Deng, H.M. Zhao, J.J. Liu, X.L. Yan, Y.Y. Li, L.F. Yin and C.H. Ding, "An improved CACO algorithm based on adaptive method and multi-variant strategies", Soft Computing, vol.19, no.3, **(2015)**,pp.701-713.

[8] M.S. Abadeh,J. Habibi, E. Soroush,"Induction of fuzzy classification systems via evolutionary ACO-based algorithms", International Journal of Simulation: Systems, Science and Technology, vol. 9, no.3, **(2008)**,pp. 1-8.

[9] V.S. Borkar and D. Das,"A novel ACO algorithm for optimization via reinforcement and initial bias", Swarm Intelligence, vol.3, no.1, **(2009)**, pp.3-34.

[10] Y. Zhang, M. Zhang and Y.C. Zhang, "A hybrid ACO/PSO algorithm and its applications", International Journal of Modelling, Identification and Control, vol.8, no.4, **(2009)**, pp. 309-316.

[11] C. Twomey, T. Stützle, M. Dorigo, M. Manfrin and M. Birattari, "An analysis of communication policies for homogeneous multi-colony ACO algorithms", Information Sciences, vol.180, **(2010)**, pp. 2390-2404.

[12] L.I. Manuel and B. Christian, "Beam-ACO for the travelling salesman problem with time windows", Computers and Operations Research, vol.37, no.9, **(2010)**, pp. 1570-1583.

[13] M.J. Meena, K.R. Chandran, A. Karthik and A. Vijay Samuel, "A parallel ACO algorithm to select terms to categorise longer documents", International International Journal of Computational Science and Engineering, vol.6, .4, **(2011)**,pp. 238-248.

[14] J.W. Zhang, S. Liu, Z. He and Z.Y. Cai, "A physical topology-aware chord model based on ACO", Journal of Computers, vol. 6,no.12, **(2011)**,pp. 2711-2718.

[15] B. Shuang,J.P. Chen, Z.B. Li, "Study on hybrid PS-ACO algorithm", Applied Intelligence, vol.34,no.1, **(2011)**, pp. 64-73.

[16] N. Medina-Rodriguez, O. Montiel-Ross, R. Sepulveda and O. Castillo, "Tool path optimization for computer Numerical control machines based on parallel ACO", Engineering Letters, vol.20, no.1, **(2012)**, pp. 101-108.

[17] T. Kötzing, F. Neumann, H. Röglin and G. Witt,"Theoretical analysis of two ACO approaches for the traveling salesman problem", Swarm Intelligence, vol.6, no.1, **(2012)**, pp. 1-21.

[18] M. Janaki Meen, K.R. Chandran, A. Karthik and A. Vijay Samuel, "An enhanced ACO algorithm to select features for text categorization and its parallelization", Expert Systems with Applications, vol.39, no.5, **(2012)**, pp. 5861-5871.

[19] A. Ugur and D. Aydin, "Improving performance of ACO algorithms using crossover mechanism based on best tours graph", International Journal of Innovative Computing, Information and Control, vol.8, no.4, **(2012)**, pp. 2789-2802.

[20] P. Guo and Z.J. Liu, "A review of improving strategies for ACO algorithms", International Journal of Digital Content Technology and its Applications, vol.6, no.16, **(2012)**, pp. 331-339.

[21] C.M. Pintea, G.C. Crisan and M. Manea, "Parallel ACO with a ring neighborhood for dynamic TSP", Journal of Information Technology Research, vol.5, no.4, **(2012)**, pp.1-13.

[22] X.G. Cheng and N.F. Xiao, "Parallel implementation of dynamic positive and negative feedback ACO with iterative", Journal of Information and Computational Science, vol.10, no.8, **(2013)**, pp. 2359-2370.

[23] L.J. Ke, Q.F Zhang and R. Battiti, "MOEA/D-ACO: A multiobjective evolutionary algorithm using decomposition and AntColony", IEEE Transactions on Cybernetics, vol.43, no.6, **(2013)**, pp. 1845-1859.

[24] R. Skinderowicz, Q.F. Zhang and R. Battiti, "Implementing population-based ACO", Lecture Notes in Computer Science, vol. 8733, **(2014)**, pp. 603-612.

[25] W. Elloumi, H. El Abed, A. Abraham and A.M. Alimi, "A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP", Applied Soft Computing Journal, vol. 25, **(2014)**, pp. 234-241.

[26] M.Z. de Souza and A.T.R. Pozo, "Parallel MOEA/D-ACO on GPU", Lecture Notes in Computer Science, vol. 8864, **(2014)**, pp.405-417.

[27] D. Jiang, "A novel hybrid optimization algorithm based on ACO and GA and its application", Recent Patents on Computer Science, vol. 8, no.2, **(2015)**,pp. 152-158.

Authors

Mengxing Li, he is an associate professor at school of Communication and Electronic Engineering, Hunan City University. He received the B.E, M.S degree from Lanzhou University in 1996, 2004 respectively, and Ph.D degree from Beijing University of Posts and Telecommunications in 2009. His research interests include artificial intelligence communication technology, signal process, robotics, *etc*.

Zhuo Wan, she is an assistant teacher at school of Communication and Electronic Engineering, Hunan City University. She received the M.S degree from Hunan University in 2015. Her research interests include optoelectronic materials, artificial intelligence communication technology, signal process, *etc*.

