Application of optimized ant colony algorithm in network routing

Bingchen Zhao*, Junying Huang, Bin Zhang, Shaofang Xia

College of mathematics and information technology, Xingtai University, Xingtai 054001, HeBei, China

Received 1 June 2014, www.cmnt.lv

Abstract

The emergence of the ant colony algorithm has caused great attention of scholars, and been widely applied in such fields as the data mining and the integrated wiring design of the large scale integrated circuits and so on. However, as the complexity to solve problems increases, the traditional ant colony algorithm increasingly shows its limitations of solving problems. Based on the ant colony optimization algorithm, this article puts forward the load-balancing routing based on the ant colony optimization, designs updating rules of pheromone concentration specific to the ant colony optimization algorithm, integrates the ant colony optimization and cross layer optimization methods, and designs updating models in terms of the volatilization of the pheromone concentration and different data groups.

Keywords: ant colony algorithm, load-balancing routing, updating rule

1 Introduction

Ant colony algorithm is the intelligent algorithm based on the ant colony by simulating the ant colony foraging behavior in the nature in the early 1990s by Italian scholars such as Dorigo. After long-term observation and research, biologists find many ants in one ant colony can realize simple and effective information exchange in a particular way, then the shortest path between the food and the ant nest can be confirmed, and this cannot be achieved in terms of one single ant. As a kind of new intelligent algorithm, the ant colony optimization algorithm is widely used in many fields in recent years.

Through the comparison with other heuristic algorithms represented by the genetic algorithm, it is not difficult for us to find that the ant colony optimization algorithm has following some characteristics: the basic idea of ant colony algorithm is the social behavior shown by imitating ants' communication relying on the pheromone. It is a kind of random general heuristics, which has very strong optimization ability. But, like similar with other optimization algorithms, as a kind of the heuristic algorithm, the ant colony algorithm itself has some shortcomings in following cases: when problem scale to be solved is large and the algorithm search reaches a certain degree, it is easy to be restrained to the partial optimal solution too early and overlook the optimal ants with shorter traversal path. It is often inconsistent with the optimal path calculated by starting from one specific point to another specific point under actual circumstances, and the solving process is relatively complicated. At the early stage, each ant search often presents certain blindness, and thus resulting in problems such as the low algorithm efficiency [1].

Based on ant colony algorithm, this article improves such problems generally existing in the ad-hoc network as the congestion problem, shortcut problem and introduced routing overhead etc. At the same time, compared with the existing load-balancing routing, the success delivery rate of the data packet, the average end-to-end delay and routing overhead are improved significantly.

2 Basic ant colony algorithm

2.1 BIOLOGICAL PROTOTYPE OF ANT COLONY ALGORITHM

Ant colony algorithm is a bionic optimization algorithm proposed in recent years, which is inspired by the real ant colony foraging behavior in the nature. Ant colony in the nature can find the shortest path from the nest to the food through mutual cooperation, and can also change with environment changes (such as sudden obstacles) to quickly find the shortest path again. Based on long-term studies by bionicist, we find: in them arching process to search for food, the ant will leave a volatile substance called pheromone along the way, and other ants can choose path according to the concentration size of such substance, and also leave this pheromones along the way to reinforce the pheromone concentration on this path, thus attracting more ants to march along this road. After a period of time, the pheromone on the shorter path is far more than the one on the longer path owing to the less volatilization and the large number of ant access at the same time. This process continues until all ant choose the shortest paths. Let us assume that there are two paths around obstacles A-B-C and A-D-C from the food source to the nest (as shown in Figure 1) [2].

*** *Corresponding author's*** e-mail: xttc2013iot@163.com

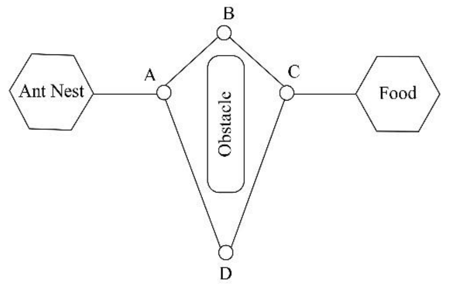

FIGURE 1 Ant foraging simulation

From the figure, we can see that since the ant can reach the food from the nest by ABC or ADC path, then the ant can also return to the nest from the food by either CBA or CDA path. Figure 1 obviously shows that ABC path is shorter than the ADC path. In the beginning, the ant has no choice but to randomly choose the path owing to there is no pheromone on the path when the ant reaches point A from the nest. From the perspective of the theory of probability, one half numbers of ants will walk by ABC, and other half by ADC path to arrive at the food source. Obviously, ants by ABC will arrive at food source first and return after taking food. Because the CBA path is shorter than the CDA path, so ants walking by CBA path will also first reach when returning to the nest from the food source. It is clear that, under the same time period and equal probability, pheromone released by ants in ABC is larger than the one in ADC, and its concentration also increases more quickly. The reason lies in that ants choose the path according to the pheromone concentration on the path: the higher the pheromone concentration is, the greater probability the path will be selected. The pheromone concentration offers basis for subsequent ants to choose the path. In this way, ants passing through ABC path will become more and more, and the pheromone will correspondingly become more and more, and the possibility for the subsequent ants to select this path will also become bigger and bigger. Soon, almost all ants will choose the ABC and CBA path to move between the nest and food source. At last, the shortest path from the nest to food source is found [3].

2.2 SYSTEMATIC CHARACTERISTICS OF ANT COLONY ALGORITHM

Artificial ants conduct the construction solution independently meanwhile at different nodes of the problem space. The solution to the whole problem will not be affected by the factor that one certain ant cannot acquire the solution successfully. It owns:

l) Diversity: the individual ant behavior is the element of the system.

2) Relevance: the inter-influence of the ant behavior.

3) Integrity: the ant colony can complete the task that the individual ant cannot complete.

The individual ant collaborates with each other to finish jointly the task finding the optimal solution. At the beginning of the algorithm, one single ant blindly constructs. After a period of the adaption, artificial ants increasingly tend to find the close optimal solution through the mutual exchange and cooperation, reflecting the self-organization process of the algorithm from the disorderliness to orderliness [4,5].

3 Improvement of basic ant colony algorithm

3.1 BASIC IDEA

First, the basic features of discrete system optimization solved by the ant colony optimization algorithm are summarized in the formalized description [6]:

1) Given the collection with limited elements $C=\{c_1,c_2,\cdots c_n\}$.

2) Definite the impossible connection / conversion limited collection L with C element in one sub-collection C^{sub} of Cartesian product

$$C\times C, L=l_{c_{ij}}\left|(c_i,c_j)\in C^{sub}, |L|\le n^2\right. .$$

3) For each element in L， definite the joint charge function $J\equiv J(l_{c_{ij}},t)$, (measure t parameterization with one certain time period).

4) Allocate limited constraint collection $\Omega\equiv\Omega(C,L,t)$ in C and L elements.

5) Definite the problem status with the sequence $s=<(c_i,c_j)>$ in C or L elements.

6) If S is the collection of all possible sequences, all feasible sub-sequence collections S^{sub} in constraint Ω (C,L,t) are the sub-collections of S. Elements in S^{sub} definite the feasible status of the problem. The length of the sequence s (i.e the element number of the sequence) is marked as $|s|$.

7) The definition of domain structure is as follows, if

a) Status S_1 and S_2 belong to S,

b) Status S_1 can come to S_2 in one logical procedure, that is, if c_1 is the ultimate element in the sequence of the confirmative status S_1, there must exist $c_2\in C$ to gain $l_{c_{ij}}\in L$, and $s\equiv<(s_1,c_1)>$, so the status S_2 is called the domain of S_1.

8) Elements in S^{sub} meeting all requirements are called one solution ψ.

9) For each solution ψ, there exist one total charge (target value) $J(L,t)$. $J(L,t)$ that are all functions $j_{c_{ij}}$ related to this solution.

Second, this section sums up several main prosperities of artificial ant colony system [7]:

1) The ant colony always searches the feasible solution with the least charge $\hat{J}_\Psi=\min_\Psi J_\Psi(L,t)$.

2) Each ant K has the memory Mk which is used to store its current path information. This memory can be

used to construct the quality of the feasible solution and appraisal solution, and the reverse path tracking.

3) The ant K in the status $S_r = < S_{r-1}, i >$ can move to any point j in the practicable domain N_i^k. Here $N_i^k = \{ j | (j \in N_i) \cap (< s_{r,j} > \in S^{sub} >) \}$.

4) Each ant will be endowed with one initial status S_s^k and one or more terminal condition e^k.

5) The ant moves from the initial status to the practicable domain status and constructs the solution in the recursive way. The construction process will end when at least one ant k meets at least one terminal condition e^k.

6) The ant k at the i point moves to the node j of N_i^k according to one certain probability decision rule.

7) When the ant moves from the point i to the domain point j, the pheromone track will be updated, and this process is called "online step-by-step pheromone update".

8) Once one solution is constructed, the ant will track reversely along the former path and update its pheromone track. This process is called "online delayed pheromone update".

After relevant basic properties are given, TSP problem is taken as an example to illustrate the basic idea and theory of the ant optimization algorithm. In terms of TSP problem, the aforementioned C is the city collection, L is the edge arc collection, and Ψ is Hamilton loop.

In the basic implementable procedures, the following variables and constants are used:
m= numbers of the ant,

$\eta_{i,j}$ = the visibility of the edge arc (i,j), that is $\frac{1}{d_{ij}}$

τ_{ij} = the pheromone track intensity of the edge arc (i,j),

$\Delta\tau_{ij}^k$ = the number of pheromone in one unit track left by the ant k on the edge arc (i,j),

Based on different τ_{ij}^k methods, different kinds of ant colony optimization algorithms are formed. The basic method is:

$$\tau_{ij}^k = \begin{cases} Q / Z_K \\ 0 \end{cases}, \tag{1}$$

P_{ij}^K =the transition probability of the ant k, which is in direct proportion to $\tau_{ij}^{\partial}\eta_{ij}^{\beta}$. J is the node that is not accessed yet.

$$\tau_{ij}^{\partial} = p\tau_{ij}^{old} + \sum_k \tau_{ij}^k . \tag{2}$$

Here, the meaning of each parameter is as follows:

α - the relative importance of the pheromone track (α≥0),

β - the relative importance of the visibility (β≥0),

ρ - the durability of the pheromone track (0≤ρ<1). 1−ρ can be understood as the evaporation of the pheromone track.

Q - it shows one constant of the pheromone track number owned by the ant.

3.2 BASIC STEPS

So, the main steps to solve TSP problem by means of the ant colony optimization algorithm can be summed as follows:

Step 1. nc←0; (nc is iterations or times to search). Put each τ_{ij} and $\Delta\tau_{ij}^k$, put m ants on n peaks;

Step 2. Put the initial starting point of each ant in the current solution collection; Move each k (k=1,....,m) to the next peak j by the probability P_{ij}^K ;Put the peak j in the current solution collection;

Step 3. Calculate the target function value of each ant Z (k=1,....,m), record the current best solution;

Step 4. Modify the track intensity based on the updating formula of the pheromone track intensity;

Step 5. In terms of each arc (τ_{ij} , τ_{ij}), put Δ←0, nc←nc+1;

Step 6. If nc is less than preset iterations and free of degeneration behavior (i.e. what found is the same solution), then repeat Step2;

Step 7. Output the current optimal solution.

Owing to the symmetry of the algorithm to the figure and the target function has no special requirement, so this algorithm can be used in all kinds of asymmetric and nonlinear problems. There is no theoretical basis for the parameters set in the algorithm, we can only determine it by experiment, and results that have been published aim at specific problems. The earliest data is acquired by solving some examples in TSP problem library-TSPLIB. Its experimental results are as follows [8]:

1) 0≤α≤5,
2) 1≤β≤5,
3) 0.1≤ρ≤0.99, ρ is optimal around 0.7,
4) 10≤Q≤10000.

4 The congestion control performance simulation of load-balancing routing agreement based on ant colony algorithm

First, the definition of each simulation data result needs to be collected:

The success delivery rate of the data packet: the data packet number reaching successfully the destination/ all data packet number generated by the modules generated from the data.

Data packet end-to-end delay (unit: second): the period of time from the data starts entering the source node of the network layer until the data is received by the network layer of the destination node.

Average end-to-end delay (unit: second): the average value of end-to-end delay of all data packets.

Routing setup delay (unit: second): from the first sending of the f ant packet from the source node to the corresponding routing response of this f ant packet received by the source node, that is, the period of time when the corresponding b ant packet is received.

The average routing setup delay (unit: second): the average value of all routing setup delays.

Routing overhead ratio: the rate of the routing control packet number(unit: packet) to all packets sent (unit: packet).

Average hop of data message: before reaching the destination node, the average frequency of the data packet that is sent and transmit equals the rate of total frequency number of data packet sent and transmit to the packet number received successfully.

We have adopted load-balancing routing protocol based on ACA, and the destination node of the data packet is randomly selected in the entire network. The specific simulation scenario parameter setting is as follows:

Network Area: 1,500m X 300 m,

Network Scale: 40 nodes,

Number of Source Nodes: 20,

Initial Distribution of Node Position: random and even distribution,

Data Rate: 1Mbps,

Communication Range: 250m,

Inter-arrival Rate of Data Packet: Poisson distribution, 3 packet each second on average,

Packet Length: 4,096 bit,

Queue Occupancy Rate Threshold (β); 0.8,

Maximum Movement Speed: 2 m/s,

Pause Time: 50, 100, 150, 200, 250, 300, 350, 400 s,

Access Control Threshold (δ); 0, 0.10, 0.15, 0.20, 0.25, 0.3,

Simulation Duration: 400s.

Figure 2, Figure 3 and Figure 4 show respectively, under different access control threshold δ, the influence of the node movement time interval exerted to the packet success delivery rate, average end-to-end delay and routing overhead. In the case that the access control threshold value δ is 100, the function of the access control rules (Ant Colony Algorithm) is almost blocked, that is the performance value of the protocol gained without using the access control rules.

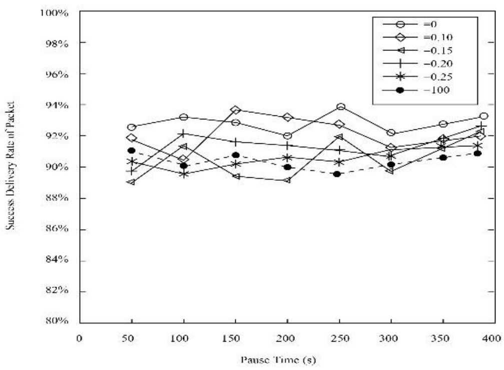

FIGURE 2 Success delivery rate of packet under different access control threshold δ

We can see from Figure 2 that the access control threshold δ exerts nearly no big effect to the success delivery rate of the packet. However, Figure 3 and Figure 4 show that the average end-to-end delay of the packet significantly decreases and the routing overhead increases with the increase of the access control threshold δ. The reason lies in that the access control becomes stricter as the access control threshold δ decreases, and the cross-layer information, such as the delay and queue occupancy rate, will be increasingly considered to determine its own path search and transmit when the routing algorithm is conducted to execute the path search and choice. That could cause the intermediate nodes to discard some path search packet according to their own situation to ease their load and also reduce the routing overhead. But the discard of the path search packet may lead, to some extent, the increase of the period of time to find the way, thus increasing the packet end-to-end delay. Therefore, it is an issue to consider how to choose the access control threshold δ to coordinate the routing overhead and end-to-end delay. At the same time, when the pause time is about 50 seconds, Figure 2, Figure 3 and Figure 4 all show the good adaptability of the access control mechanism to the mobility.

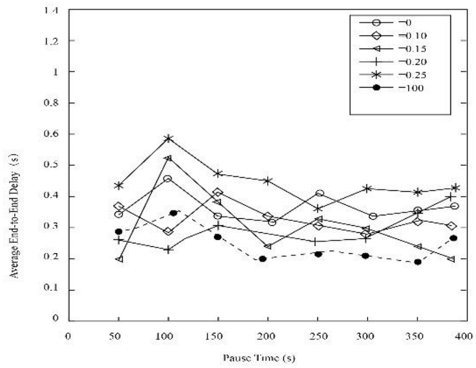

FIGURE 3 Average end-to-end delay under different access control threshold δ

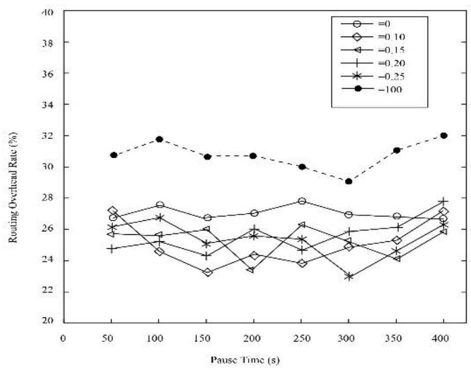

FIGURE 4 Routing overhead rate under different access control threshold δ

5 Conclusion

The experimental results show that the above work we have done in the field of ant colony algorithm greatly improves the performance of basic ant colony algorithm, accelerates the convergence speed, improves the quality of the understanding and expands the scope of its application. Of course, there exists some deficiency that needs to be further studied and discussed in this article. For example, the effect of the shared information in the cross-layer design to the loading balance needs further research, and how to effectively use the shared information and how to realize the dynamic perceptive information with the purpose to realize the dynamic self-adaptation and performance optimization of nodes in this system. Meanwhile, the setting of each parameter in the updating formulas of the pheromone concentration in the system will be a topic for research.

Acknowledgments

This work was supported by Science and Technology Department of Hebei Province Self-financing project (13270134).

References

[1] Forsati R, Moayedikia A 2014 Enriched ant colony optimization and its application in feature selection *Neuro computing* **142**(22) 354-71

[2] Savsani P, Jhala R L 2013 Effect of hybridizing Biogeography-Based Optimization (BBO) technique with Artificial Immune Algorithm (AIA) and Ant Colony Optimization (ACO) *Applied Soft Computing* **21**(8) 542-53

[3] Fingler H, Cáceres E N 2012 A CUDA based Solution to the Multidimensional Knapsack Problem Using the Ant Colony Optimization *Procedia Computer Science* 29 84-94

[4] He Z, Qi H 2014 Inverse estimation of the spheroidal particle size distribution using Ant Colony Optimization algorithms in multispectral extinction technique *Optics Communications* **328**(1) 8-22

[5] Huang X, Zou X 2014 Measurement of total anthocyanins content in flowering tea using near infrared spectroscopy combined with ant colony optimization models *Food Chemistry* **164**(1) 536-43

[6] He Y-J, Ma Z-F 2013 Optimal design of linear sensor networks for process plants: A multi-objective ant colony optimization approach *Chemometrics and Intelligent Laboratory Systems* **135**(15) 37-47

[7] Sina T, Moradi P 2014 An unsupervised feature selection algorithm based on ant colony optimization *Engineering Applications of Artificial Intelligence* **32**(6) 112-23

[8] Liao T, Stützle T 2013 A unified ant colony optimization algorithm for continuous optimization *European Journal of Operational Research* **234**(3) 597-609

Authors

Bingchen Zhao, 30.09.1976, China.

Current position, grades: teacher at Xingtai University, China.
University studies: Master's degree in Electronic and communication from China University of Mining and Technology, China in 2008.
Scientific interest: embedded computer and internet of things.

Junying Huang, 19.10.1983, China.

Current position, grades: teacher at Xingtai University, China.
University studies: Master's degree in Computer Application Technology from Liaoning Normal University, China in 2008.
Scientific interest: embedded computer and computer graphics.

Bin Zhang, 07.09.1981, China.

Current position, grades: College of mathematics and information technology, Xingtai University.
University studies: Master's degree master's degree of electronic circuit and system in Beijing University of Technology in 2010.
Scientific interest: embedded system.

Shaofang Xia, 27.01.1983, China.

Current position, grades: teacher at Xingtai University, China.
University studies: Master's degree in Computer Application Technology from Taiyuan University of Science and Technology, China in 2008.
Scientific interest: embedded computer and internet of things.