# An Improved Ant Colony Algorithm for Probabilistic QoS Routing

Genhong Ding, Dongwei Guo
College of Sciences
Hohai University
Nanjing, China
dgh@hhu.edu.cn, 525562241@qq.com

Yuchen Ding
Belk College of Business
University of North Carolina at Charlotte
Charlotte, USA
yding6@uncc.edu

*Abstract*—**The actual dynamic network environment urgently requires us to give full consideration to the non-precision of the network state when we design the QoS routing algorithm. Mainly by substituting the piecewise function for the probability constant which is chosen by ants when a route is selected, changing pheromone update rule and introducing definition of the ant age, this paper improves the ant colony algorithm for probabilistic QoS routing. Under the case study, the simulation shows that the improved algorithm can find the optimal solution 100% if the case has the best one. This result is much better than the one obtained by the basic ant colony algorithm. The improved algorithm can solve the probabilistic QoS network routing problem effectively.**

***Keywords-ant colony algorithm; probability; QoS; routing algorithm***

## I. INTRODUCTION

With the development of the Internet, how to make full use of network resources to meet the diverse QoS requirements becomes very important. Nowadays scholars have put forward a number of helpful models to solve the problem of QoS, such as protocols, strategies, algorithms, and scheduling mechanism. Fariza Sabrina [1] presented a novel QoS-aware resource scheduling algorithm called Weighted Composite Bandwidth and CPU Scheduler, which jointly allocates the fair share of the link bandwidth as well as processing resource to all competing flows. Doulamis et al [2] used least square algorithm to predict task work load, and used this information to obtain better results for resource scheduling in grid computing. QoS routing is the key technology to guarantee the quality of services. The key to solving the problem of QoS routing is to find a route to meet the QoS routing constraints in order to make the optimal allocation of network resources. In recent years, many scholars used the genetic algorithm [3], particle swarm optimization algorithm [4] and other heuristic intelligent algorithm to solve this problem, and obtained good results.

Ant colony algorithm has great advantages in solving complicated optimization problems; however it has some disadvantages, such as that it is easy to fall into local optimal solution [5]. This paper makes some relevant improvements for basic ant colony algorithm, builds the probabilistic network model, and proposes Improved Ant Colony Algorithm for probabilistic QoS Routing. The simulation proves the effectiveness of the algorithm which is used to solve the network routing problem.

## II. ANT COLONY ALGORITHM AND ITS IMPROVEMENTS

### A. *The Mathematical Model of Basic Ant Colony Algorithm*

Basic Ant Colony Algorithm (BACA) is a kind of heuristic algorithms [6], which simulates the behavior of ants foraging. The TSP problem can be solved by ant colony algorithm. Here is the mathematical model of the problem. $N=\{1,2,\cdots,n\}$ is defined as the set of the cities, $d_{ij}(i,j=1,2,\cdots,n)$ the distance between the city $i$ and the city $j$, $m$ the total number of ants, $b_i(t)$ the total number of ants that exist in the city $i$ at time $t$, $\tau_{ij}(t)$ the amount of pheromone available in the *route*$(i, j)$ at time $t$, and $\tau_{ij}(0)=C$ ($C$ is a constant). Every ant has the following features:

*1)* The Ants choose the city in accordance with a certain probability based on the pheromone intensity on route.

*2)* The cities selected by ant $k$ are not been visited by ant $k$ before, the allowable set is denoted by $allowed_k$, and the city walked through by ant $k$ is recorded by taboo table $tabu_k$.

*3)* After completing a visit, ants leave pheromone in each edge which they have visited.

Ant $k$ shifts the direction according to the amount of information on each edge. $p_{ij}^k(t)$ is denoted as the probability for ant $k$ Shifting from city $i$ to city $j$ at time $t$.

$$p_{ij}^k=\begin{cases}\dfrac{[\tau_{ij}(t)]^\alpha[\eta_{ij}(t)]^\beta}{\sum\limits_{s\in allowed_k}[\tau_{is}(t)]^\alpha[\eta_{is}(t)]^\beta}, if & j\in allowed_k\\ 0, & otherwise\end{cases} \tag{2.1}$$

Where $\eta_{ij}$ represents pre-specified heuristic information. We set $\eta_{ij}=1/d_{ij}$. We use $\alpha$ and $\beta$ to represent the relative importance of the amount of pheromone and the predictable information respectively.

As time goes, the pheromone left by ants will continue to evaporate in each route. We denote $\rho$ as the degree of evaporation of pheromone, and $0<\rho\le 1$. After the ants complete the traversal of $n$ cities, the amount of information on the route should be adjusted according to the following rules at time $t$+1:

$$\tau_{ij}(t+1)=(1-\rho)\tau_{ij}(t)+\rho\Delta\tau_{ij}(t) \tag{2.2}$$

$$\Delta\tau_{ij}(t)=\sum_{i=1}^{m}\Delta\tau_{ij}^k(t) \tag{2.3}$$

The pheromone updating strategy is as follows:

$$\Delta\tau_{ij}^{k}(t)=\begin{cases}\dfrac{Q}{L_k}, & the\ ant\ k\ pass(i,j)\ in\ this\ cycle\\ 0, & otherwise\end{cases} \tag{2.4}$$

Where $\Delta\tau_{ij}^{k}$ represents the amount of pheromone that ant $k$ left in this cycle on route $(i, j)$. $\Delta\tau_{ij}$ represents the incremental amount of pheromone in this cycle on route $(i, j)$, $Q$ represents the total amount of pheromone and $L_k$ represents the length of the cycle walked by ant $k$.

## B. Improved Ant Colony Algorithm (IACA)

### 1) The improvements of Ant Colony Algorithm

Ant colony algorithm is improved by many scholars and many results have been achieved. By adjusting heuristic function, changing pheromone update rule and introducing fitness function, Yu Feng et al [5] improved search efficiency for the basic ant colony algorithm. Cao Huaihu [7] proposed a QoS routing algorithm based on mobile agent and ant colony, which combines various constraints and network load conditions with the ant colony algorithm in the pheromone. For the characteristic that the ant colony algorithm is easy to fall into local optimal solutions, we make the following improvements to the ant colony algorithm:

*a)* Use the piecewise function $q_0(t)$ to replace the constant $q_0$ chosen by ants when selecting the route. In the very beginning of the iteration, $q_0(t)$ is relatively small, so that ants can have a greater probability to explore other solutions. In the later, we can choose a relatively large $q_0(t)$ to speed up the convergence

*b)* For the algorithm to escape from the local optimal solution, this paper uses $Q(t)$ instead of $Q$ of (2.4). It should meet the following conditions:

*(i)* In the very beginning of the iteration, the value of $Q(t)$ should be smaller, so that the increment of pheromone in each route will be small to avoid too strong positive feedback.

*(ii)* In the later iteration, the value of $Q(t)$ become bigger to speed up the convergence. Simulation results show that smooth functions are batter than piecewise functions on the operation effect. To sum up the above conditions, in this paper, we set $Q(t)=\log(t+1)$.

*c)* Due to the positive feedback, the amount of pheromone often goes to extremes in some local optimal area. It is better to set range for pheromone. For $\forall\tau_{ij}(t)$

$$Pheromone\_\min \le \tau_{ij}(t) \le Pheromone\_\max, i, j\in\{1,\cdots,n\} \tag{2.5}$$

If out of range, the boundary value will be taken.

*d)* As the ant colony algorithm is a stochastic search algorithm, ants are endowed with the concept of age to speed up the global convergence. The age of artificial ants is the hops, which are the node number that the ants elapsed in the network. When an ant reaches the prescriptive upper age limit and has not yet found destination node, the ant will die.

### 2) The design of algorithm

The rules of the improved algorithm are as follows:

*a) The rules of state transition*

According to the following rules, Ant $k$ in node $i$ selects the next node $j$ at time $t$:

$$j=\begin{cases}\arg\max\limits_{s\in allowed_k}\left\{\tau_{is}(t)^{\alpha}\eta_{is}(t)^{\beta}\right\}, & if\ q\le q_0(t)\\ choose\ j\ according\ to(2.7), & otherwise\end{cases} \tag{2.6}$$

$$p_{ij}^{k}=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum\limits_{s\in allowed_k}[\tau_{is}(t)]^{\alpha}[\eta_{is}(t)]^{\beta}}, & if\ j\in allowed_k\\ 0 & , otherwise\end{cases} \tag{2.7}$$

$$q_0(t)=\begin{cases}q_1, & t\le T1\\ q_2, & T1<t\le T2\\ q_3, & t>T2\end{cases} \tag{2.8}$$

Where $q$ is a uniform random number in the interval [0, 1]. $q_1, q_2, q_3$ are constants, and $q_1<q_2<q_3$. $\eta_{ij}(t)$ is a heuristic function, here equals the delay between node $i$ and node $j$.

*b) The local update rule for pheromone*

When ant $k$ passes the route $(i, j)$, the pheromone of the route will be updated according to the formula (2.9).

$$\tau_{ij}(t)=(1-\rho_0)\cdot\tau_{ij}(t)+\rho_0\cdot cons \tag{2.9}$$

Where $0<\rho_0\le 1$, it shows the speed of the old pheromone evaporation in the route.

*c) The global update rule for pheromone*

After all ants complete a search successfully, we choose the global best ant in the current iteration.

$$\tau_{ij}(t+1)=(1-\rho_1)\cdot\tau_{ij}(t)+\rho_1\cdot\Delta\tau_{ij}(t) \tag{2.10}$$

and

$$\Delta\tau_{ij}=\begin{cases}Q(t)/L_{best}, & e_{ij}\in current\ best\ route\\ 0 & , otherwise\end{cases} \tag{2.11}$$

$$Q(t)=\log(t+1) \tag{2.12}$$

Where $\rho_1$ represents the intensity of evaporation, in this paper $\rho_1=\rho_0$. $L_{best}$ represents the sum of the cost of the optimal routing in the current cycle.

# III. IACA APPLICATIONS IN QOS ROUTING

## A. Probabilistic Network Model

The Probabilistic network model can be described as a weighted graph $G(V,E)$, where $V=\{v_1,v_2,\cdots,v_n\}$ represents the set of nodes, and $E=\{e_1,e_2,\cdots e_e\}$ represents the of links. In this paper, only the cost, the bandwidth and the delay are considered for IACA to find Probabilistic QoS Routing. Therefore, each link can be assigned the following triple:

$$[\cos t,(bandwidth_{\min},bandwidth_{\max}),(delay_{\min},delay_{\max})]$$

Where, the element of the formula represents the cost of the link, the range of bandwidth and the range of link delay respectively.

For a routing request $(s,d)$, according to requirements of the QoS, the purpose of the routing algorithm is to find a minimum cost rout that meets the following two conditions:

$$PR(bandwidth(P(s,d)) \geq B_{\min}) \geq f_B \quad (3.1)$$

$$PR\left(\sum_{(a,b)\in P(s,d)} delay_i(a,b) \leq D_{\max}\right) \geq f_D \quad (3.2)$$

Where, $PR(A)$ represents the probability of occurrence of the event $A$. $delay_i(a,b)$ is the actual delay of the link $(a,b)$, and link $(a,b)$ is the $i$th section of route $P(s,d)$. $B_{\min}$ represents the minimum bandwidth required by the application. $D_{\max}$ represents the maximum delay required by the application. $f_B$ represents the minimum allowed probability which satisfies the bandwidth constraints of $P(s,d)$. $f_D$ represents the minimum allowed probability which satisfies the delay constraints of $P(s,d)$.

If the available bandwidth is imprecise, we suppose $bandwidth(a,b)$ is a random variable which has uniform distribution in the interval $[bandwidth(a,b)_{\min}$, $bandwidth(a,b)_{\max}]$. We can get the probability of a given route $P(s,d)$ which meets the bandwidth requirements:

$$PR(bandwidth(P(s,d)) \geq B_{\min}) = \prod_{(a,b)\in P(s,d)} \min\left(\frac{bandwidth(a,b)_{\max} - B_{\min}}{bandwidth(a,b)_{\max} - bandwidth(a,b)_{\min}}, 1\right) \quad (3.3)$$

Obviously, if $\exists (a,b) \in P(s,d)$, $B_{\min} \geq bandwidth(a,b)_{\max}$ holds, then $PR(bandwidth(P(s,d)) \geq B_{\min}) = 0$. So we can consider deleting the link whose maximum possible bandwidth is less than $B_{\min}$ when we design the algorithm, which can improve the calculating efficiency.

In the case that the delay information of the link is imprecise and the delay can be added, we denote $\sum_{(a,b)\in P(s,d)} delay_i(a,b)$ as the total delay of the route $P(s,d)$, in short $D$. We suppose that random variable $delay_i(a,b)$ has normal distribution $N(\mu_i, \sigma_i^2)$. According to the principle of $3\sigma$, we change the distribution interval $(-\infty, +\infty)$ of random variable $delay_i(a,b)$ to $(\mu_i - 3\sigma_i, \mu_i + 3\sigma_i)$. Then

$$\mu_i = \frac{delay_i(a,b)_{\min} + delay_i(a,b)_{\max}}{2} \quad (3.4)$$

$$\sigma_i = \frac{delay_i(a,b)_{\max} - delay_i(a,b)_{\min}}{6} \quad (3.5)$$

According to the property of normal distribution, the total delay $D$ of the route $P(s,d)$ has normal distribution $N(\mu, \sigma^2)$. Where $\mu = \sum \mu_i$, and $\sigma^2 = \sum \sigma_i^2$. The probability of the total delay of route $P(s,d)$ which meets the delay requirements is as follows:

$$PR(D \leq D_{\max}) = \begin{cases} 0 & , D_{\max} < \mu - 3\sigma \\ \int_{\mu-3\sigma}^{D_{\max}} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} dx & , \mu - 3\sigma \leq D_{\max} \leq \mu + 3\sigma \\ 1 & , D_{\max} > \mu + 3\sigma \end{cases} \quad (3.6)$$

*B. The Implementation of the Algorithm*

The steps of the algorithm are as follows:

*1)* Initialization: Set $[\cos t, (bandwidth_{\min}, bandwidth_{\max}), (delay_{\min}, delay_{\max})]$ for each edge. Set source node $s$ and destination node $d$. The initial pheromone intensity $\tau_{ij}(0) = C$, and $\Delta\tau_{ij}(0) = 0$. Set taboo table $tabu_k$ for ant $k$, which is used to record the nodes that has been traversed. $\max\_hop$ is the maximum number of hops that ants can travel. $\max\_iteration$ is the maximum number of iterations. $t$ is the number of iterations. Set $t = 1$, and delete the edge whose maximum possible bandwidth is less than $B_{\min}$.

*2)* Put $m$ ants on the source node and add the index of the source node to the ant's taboo table. Set $hop_k = 0$, where $hop_k$ is the hops of ant $k$. The ants start from the source node. Set $i = s$.

*3)* If $hop_k \geq \max\_hop$, then ant $k$ dies; Otherwise, ant $k$ will starts from the node $i$. Select the next node $j$ from the candidate set of $allowed_k$ according the transfer rules of (2.6) and (2.7), and set $hop_k = hop_k + 1$. Update the pheromone on edge $(i,j)$ by (2.9) and check if it satisfies (2.5). If $j = d$, ant $k$ stops going to the next step.

*4)* For ant $k$, if $PR(bandwidth(P(s,j)) \geq B_{\min}) < f_B$, then ant $k$ dies.

*5)* Repeat until all ants complete step *3)*.

*6)* Calculate the cost of the route in which the ants travel and the probability of satisfying the bandwidth constraint and the delay constraint. Find out the minimum cost route that meets the condition of $PR(D \leq D_{\max}) \geq f_D$. Update the global pheromone by using (2.10), and test (2.5).

*7)* If $t < \max\_hop$, then set $t = t + 1$, and go to *2)*; otherwise, output the optimal route, and stop the program.

*C. Simulations and Algorithm Analysis*

In order to test the performance of IACA for probabilistic QoS routing, we set the following experimental parameters: the number of ants $m$ = 30, the number of iterations max_*iteration* = 100, the initial strength of pheromone $\tau_{ij}(0) = 0.000001$, the evaporation intensity of the pheromone in local update $\rho_0$ = 0.1, also in global update $\rho$ = 0.1.The given QoS requirements are as follows: $B_{\min} = 50$, $D_{\max} = 80$, $f_B = 0.55$, $f_D = 0.5$. We use 10 unicast routing request for simulations, and run BACA and IACA 300 times for each request respectively and record the number of the times when optimal solutions are found, including the corresponding percentage. The network structure used for cases study is shown below.

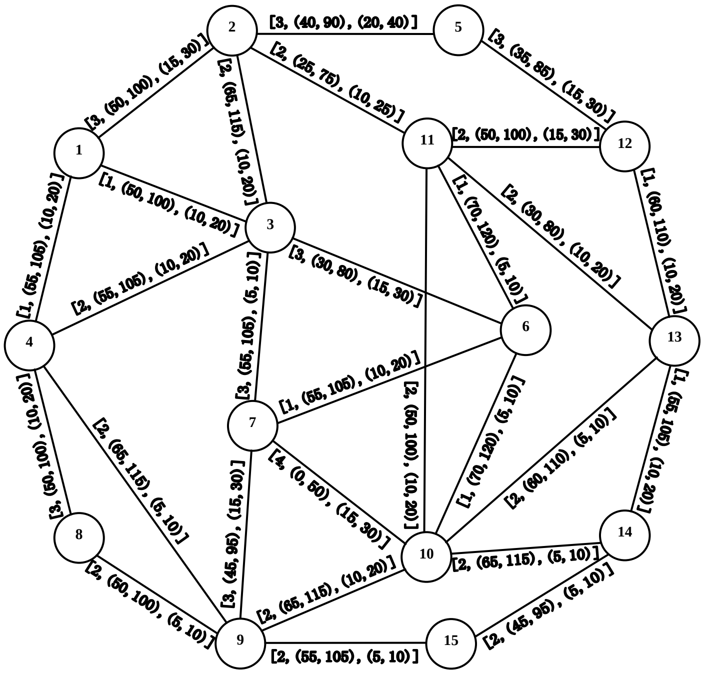


Figure 1. Network structure

The triple in Fig. 1 represents $[\cos t, (bandwidth_{\min}, bandwidth_{\max}), (delay_{\min}, delay_{\max})]$ for each edge. The simulation results shown in Table 1.

When the bandwidth constraint meets the uniform distribution and the delay time constraint meets the normal distribution, it can be seen from the simulation that the rate of optimal solutions found by BACA can reach 86.5%, while IACA can reach 100%. Therefore, IACA can find a variety of optimal routing schemes steady and easily.

## IV. CONCLUSIONS

Traditional QoS routing algorithms are all assumed that the whole network states are precise. However, the actual dynamic network environment urgently requires us to give full consideration to the non-precision of the network states when the QoS routing algorithm is designed. This paper mainly uses the piecewise function to replace the probability constant chosen by ants when a route is selected, changes pheromone update rule and gives the concept of the ants age to improve the basic ant colony algorithm. We build the network model which is based on imprecise state information and put forward the corresponding QoS routing algorithm based on the improved ant colony algorithm. The simulation results show that the algorithm can solve the probabilistic QoS routing problem effectively. Using IACA to find optimal solutions is easier than using BACA.

TABLE I. THE COMPARISON OF THE RESULTS BETWEEN BACA AND IACA

| **Route request** | **Optimal route selected by the algorithms** | **Cost** | **The probability satisfying the bandwidth demand** | **The probability satisfying the delay demand** | **Number of times for optimal solutions found (and percentage)** | |
|---|---|---|---|---|---|---|
| | | | | | ***BACA*** | ***IACA*** |
| (1,13) | 1→3→6→10→13 | 7 | 0.6 | 0.554562 | 45 (15%) | 60 (20%) |
| | 1→4→9→10→13 | 7 | 1 | 0.681687 | 213 (71%) | 240 (80%) |
| (2,14) | 2→3→6→10→14 | 8 | 0.6 | 0.554562 | 266 (88.7%) | 300 (100%) |
| (4,12) | 4→9→10→13→12 | 7 | 1 | 0.997587 | 249 (83%) | 300 (100%) |
| (4,14) | 4→9→10→14 | 6 | 1 | 1 | 2 (0.7%) | 5 (1.7%) |
| | 4→9→15→14 | 6 | 0.9 | 1 | 271 (90.3%) | 295 (98.3%) |
| (9,12) | 9→10→13→12 | 5 | 1 | 0.9987 | 264 (88%) | 300 (100%) |
| (5,9) | 5→12→13→10→9 | 8 | 0.7 | 0.503256 | 252 (84%) | 300 (100%) |
| (5,15) | 5→12→13→14→15 | 7 | 0.63 | 0.50325 | 261 (87%) | 300 (100%) |
| (4,13) | 4→9→10→13 | 6 | 1 | 1 | 273 (91%) | 300 (100%) |
| (1,14) | 1→4→9→15→14 | 7 | 0.9 | 0.722758 | 237 (79%) | 295(98.3%) |
| | 1→4→9→10→14 | 7 | 1 | 0.681687 | 3 (1%) | 4 (1.4%) |
| | 1→3→6→10→14 | 7 | 0.6 | 0.554562 | 0 (0%) | 1 (0.3%) |
| (8,12) | 8→9→10→13→12 | 7 | 1 | 0.997587 | 259 (86.3%) | 300 (100%) |
| The sum of the result of 10 route requests | | | | | 2595(86.5%) | 3000 (100%) |

## REFERENCES


[1] Fariza Sabrina, "A novel resource scheduling algorithm for QoS-aware services on the Internet," Computers and Electrical Engineering, 2010, 36, pp. 718-734.

[2] Nikolaos Doulamis, Anastasios Doulamis, Antonios Litke, Athanasios Panagakis, Theodora Varvarigou, and Emmanuel Varvarigos, "Adjusted fair scheduling and non-linear workload prediction for QoS guarantees in grid computing," Computer Communications, 2007,30, pp. 499-515.

[3] R. Leela, N. Thanulekshmi, and S. Selvakumar, "Multi-constraint Qos Unicast Routing Using Genetic Algorithm," Applied Soft Computing, 2011, 11(2), pp. 1753-1761.

[4] Rehab F. Abdel-Kader, "Hybrid discrete PSO with GA operators for efficient QoS-multicast routing," Ain Shams Engineering Journal, 2011, 2(1), pp.21-31.

[5] Yu Feng and Bai Liang, "Study of QoS Routing Based on Improved Ant Colony Algorithm," Journal of Shenyang Jianzhu University (Natural Science), 2011, 27(4), pp.782-787.

[6] Dorigo M, Gambardella L M., "Ant colony system: a cooperative learning approach to the traveling salesman problem," IEEE Transactions on Evolutionary Computation, 1997, 1(1), pp.53-66.

[7] Cao Huaihu, "A QoS Routing Algorithm Based on Ant Colony Optimization and Mobile Agent," Procedia Engineering, 2012, 29, pp. 1208-1212.