

# A Quality of Service Anycast Routing Algorithm Based on Improved Ant Colony Optimization

Yongsheng Li<sup>1,2</sup>

<sup>1</sup>School of Information Science and Engineering, Guangxi University for Nationalities, Nanning, China

<sup>2</sup>Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Nanning, China

Email: lyshlh@163.com

**Abstract**—Quality of Service (QoS) anycast routing problem is a nonlinear combination optimization problem, which is proved to be a NP-complete problem, at present, the problem can be prevalently solved by heuristic methods. Ant colony optimization algorithm (ACO) is a novel random search algorithm. On the one hand, it does not depend on the specific mathematical description, on the other hand, which has the advantages of robust, positive feedback, distributed computing. Consequently, ACO has been widely used in solving combinatorial optimization problems. However, the basic ACO has several shortcomings that the convergence rate is slow and it's easily to stuck in local optimum for solving QoS anycast routing problem. In this paper, the basic ACO has been improved, firstly, iteration operator is introduced in the node selection, which can make the node selection strategy is adjusted dynamically with the iteration. Secondly, pheromone evaporation coefficient is adjusted adaptively according to the distribution of ant colony. Finally, according to the evolutionary speed of the population, the premature convergence is estimated. The mutation and secondary ant colony operation is introduced, which can make the algorithm successfully to escape from local optima, and can rapidly approximate to the global optimum. Simulation results show that the algorithm has preferable global search ability and can effectively jump out of local optimum and rapidly converge to the global optimal solution. Thereby, the algorithm is feasible and effective.

**Index Terms**—Ant Colony Algorithm, QoS Anycast Routing, iteration operator, dispersed degree, populations of the speed of evolution.

## I. INTRODUCTION

With the rapid growth of Internet, more and more consumers get all kinds of information by Internet. In order to meet the needs of consumers, the information sites in internet must provide faster service with better communication quality of service. A basic technique is that to use simple replication or mirroring. However, this method is not transparent for the user, thus the choice is blindfold. Choosing one of many functionally identical machines has been and will continue to be a major issue as

the internet scales up. Anycast is a new communication service and which is defined in IPV6 [1]. It allows a host to communicate with the “nearest” destination host, which is one of the destination hosts that have the same anycast address and provide the same service. In addition, in order to provide better server selection and load balancing, anycast supports auto-configuration in DNS and has important application in ad hoc networks and sensor networks[2]. With the rapid expansion of the Internet, which leads to the usage of replicated servers and other anycast services are increasing, an effective anycast routing algorithms are becoming important to solve this problem. The anycast routing problem (ARP) of finding a optimum path from a source node to any member of destination group with multiple QoS requirements, is a nonlinear combination optimization problem, which is proved to be a NP complete problem[3,4]. We can't get satisfying results when we use the precise method in polynomial time. At present, many heuristic methods are presented for anycast routing problem and some research results has been recognized. In [5], an improved particle swarm optimization algorithm for QoS anycast routing is proposed. The algorithm uses a special add operator to make the worst path learning rom the better path in order to approach to global optimal path and a random mutation operator is designed to mutate global optimal randomly. In [6], a QoS anycast routing algorithm based on immune genetic algorithm is proposed and the algorithm combines the immune algorithm with the traditional genetic algorithm, and it reserves the superior search ability of original algorithm for global search and avoids poor performance of local search and precocious phenomenon. The works of [7] is to adopt the method of Depth\_First Search with weight and roulette method to make the original colony multiple. On the same time, a method of instructional aberrance is brought forward. An anycast routing algorithm based on simplified particle swarm optimization and diversity strategy is proposed in [8]. Firstly, the algorithm simplified the complexity of standard PSO algorithm and improved the convergence rate extraordinarily by dynamically changing the inertia weight. Then, a mutation operator and diversity strategy was employed to maintain particle's diversity; sequentially, global and local search effectively can be balanced. Finally, the validity and efficiency of the

Manuscript received May 20, 2012; revised August 17, 2012; accepted August 30, 2012.

Project numbers: Natural Science Foundation of Guangxi Province of China (No. 2010GXNSFB013052) and Youth Foundation of Guagnxi University for Nationalities of China (No. 2011MDQN040).

existing algorithm are demonstrated by network experiment. These algorithms are too complex and easy to fall into local optima and their convergence rate is slow for solving multiple QoS constrained anycast routing problems.

Ant colony optimization (ACO) is a meta-heuristic algorithm for hard discrete optimization problems that was first proposed in the early 1990s [9-11]. The ant colony algorithm has many distinguished merits such as positive feedback, distributed computing and constructive greedy heuristic searching. ACO has been a fruitful paradigm for designing effective combinatorial optimization solution algorithms which has been successfully applied to many complex optimization problems, such as the travelling salesman problem (TSP), the quadratic assignment problem (QAP), vehicle routing, the Job Shop Scheduling Problem (JSP)[12], etc.

This paper organizes as follows. In Section II , a formal definition of the anycast routing with multiple QoS constraints is introduced. The principle of ant colony algorithm is introduced in Section III. In Section IV, an improved ACO algorithm for QoS anycast routing is presented. Experiments are done in Section V and some concluding remarks are given in Section VI.

## II. QoS ANYCAST ROUTING PROBLEM FORMULATION

A network can be represented as an undirected weighted graph  $G(V, E)$  [13-16], where  $V = \{v_1, v_2 \dots v_n\}$  is the set of the network nodes and  $E = \{e_1, e_2 \dots e_m\}$  is the edges of the network.  $s \in V$  denote the source host that may send packets with anycast address. Given an anycast packet with anycast(destination) address  $M$  ,  $M \in \{V - \{s\}\}$  denote the group of designated recipients. For each  $e \in E$  , which has four parameters, namely bandwidth function:  $bandwidth(e)$ , delay function : $delay(e)$ , delay jitter function:  $delay\_jitter(e)$ , cost function:  $cost(e)$ . For each  $n \in V$  , it also has four parameters, namely delay function:  $delay(n)$ , packet loss function:  $packet\_loss(n)$ , delay jitter function:  $delay\_jitter(n)$  , cost function :  $cost(n)$ .

In QoS metric parameters, bandwidth is protuberance metric parameter, delay, cost and delay jitter are addable metric parameters, but packet loss rate is multiple metric parameter. Then for the given source host  $s \in V$  and the group of designated recipients  $M$  , the anycast(destination) address  $t$  ( $t \in M$ ), the path  $p(s, t)$  and the functions of bandwidth, delay, packet loss, delay jitter, cost can be formulated as followings:

$$bandwidth(p(s, t)) = \min(bandwidth(e)) \quad (1)$$

$$delay(p(s, t)) = \sum_{e \in p(s, t)} delay(e) + \sum_{n \in p(s, t)} delay(n) \quad (2)$$

$$packet\_loss(p(s, t)) = 1 - \prod_{n \in p(s, t)} (1 - packet\_loss(n)) \quad (3)$$

$$delay\_jitter(p(s, t)) = \sum_{e \in p(s, t)} delay\_jitter(e) + \sum_{n \in p(s, t)} delay\_jitter(n) \quad (4)$$

$$cost(p(s, t)) = \sum_{e \in p(s, t)} cost(e) + \sum_{n \in p(s, t)} cost(n) \quad (5)$$

The path  $p(s, t)$  is the feasible path from the source node  $s$  to the anycast(destination) address  $t$ . The aim of QoS anycast routing is finding the path  $p(s, t)$  which must simultaneous satisfy following conditions:

(1) delay constraint:

$$delay(p(s, t)) \leq D \quad (6)$$

(2) bandwidth constraint:

$$bandwidth(p(s, t)) \geq B \quad (7)$$

(3) delay jitter constraint:

$$delay\_jitter(p(s, t)) \leq DJ \quad (8)$$

(4) packet loss constraint:

$$packet\_loss(p(s, t)) \leq PL \quad (9)$$

(5) cost constraint:

$$\min(Cost(p(s, t))) \quad (10)$$

In all feasible path from the source node  $s$  to the anycast(destination) address  $t$  , the cost of  $p(s, t)$  is minimal.

Where,  $D$  is the delay constraint,  $B$  is the bandwidth constraint,  $DJ$  delay jitter constraint,  $PL$  is the packet loss constraint.

## III. THE PRINCIPLE OF ANT COLONY OPTIMIZATION ALGORITHM

The ant colony optimization algorithm (ACO) is proposed in the early 1990s [9-11]. In 1996, ant colony system is proposed by Dorigo and Gmabardella[11,17-19] , the performance of the ACO is effectively improved. In [11], they made three improvements as follows:

(1) A new selection strategy that combination of deterministic selection and random selection is adapted, which both can utilize the advantage of prior knowledge and can tententiously explore. For an ant at node  $r$  to move to the next city  $s$ , the state transition rule is given by the following formula.

$$s_k = \begin{cases} \arg \max_{u \in allowed_k} \{[\tau(r, u)]^\alpha [\eta(r, u)]^\beta\} , q \leq q_0 \\ S \quad q > q_0 \end{cases} \quad (11)$$

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{i,j}^\alpha(t) \eta_{i,j}^\beta(t)}{\sum_{s \in allowed_k} (\tau_{i,s}^\alpha(t) \cdot \eta_{i,s}^\beta(t))} , j \in allowed_k \\ 0 \quad otherwise \end{cases} \quad (12)$$

Where  $s_k$  is the next node of ant  $k$ ,  $q$  is the random number draw from  $[0,1]$ ,  $q_0$  is a parameter ( $0 \leq q_0 \leq 1$ ).  $S$  is a random variable selected by the probability distribution given in (12).  $allowed_k$  is a node set that these node can be selected for ant  $k$  in the next time,  $\alpha$  is the pheromone heuristic factor, which reflects the effect of pheromone by ant accumulates when ants move to the other nodes.  $\beta$  is a heuristic factor, had reflects the degree of the heuristic information is focused when the ants select path.  $\tau_{ij}(t)$  is the pheromone of path  $(i, j)$  at  $t$  time.  $\eta_{ij}$  is the visibility of path  $(i, j)$ , which is corresponding with the inverse of the distance from node  $i$  to node  $j$ .

(2) Only the global optimal ant path performs global updating rule. After each iteration, the pheromone is enhanced only occur on the path walked by the best ant. For other pathway, the pheromone will be gradually reduced due to volatile mechanism, which can make the ant colony more inclined to select optimal path. Consequently, the convergence rate will be increased and the search efficiency will be enhanced. Global update rule is described as follows:

$$\tau(r, s) \leftarrow (1 - \rho) \cdot \tau(r, s) + \rho \cdot \Delta \tau(r, s) \quad (13)$$

$$\Delta \tau(r, s) = \begin{cases} Q / L_{gb} & \text{if } (r, s) \in g \\ 0 & \text{else} \end{cases} \quad (14)$$

Here  $\rho$  is the pheromone volatile coefficient,  $0 < \rho < 1$ ,  $L_{gb}$  is the current global optimal path.  $Q$  is a constant that indicate initial pheromone intensity between two nodes.

(3) Using the local update rule. The pheromone will be local updated when these ants build a path, which make the pheromone release by ants to reduce when they pass the path. The local update rule is used to decrease influence on other ants and make them search other edges. Therefore, the ants can avoid that they prematurely converge to a same solution.

The local update rule is represented by (15).

$$\tau(r, s) \leftarrow (1 - \rho_0) \cdot \tau(r, s) + \rho_0 \cdot \tau_0 \quad (15)$$

$$\tau_0 = (nL_{nm})^{-1} \quad (16)$$

Where  $n$  is the number of nodes.  $L_{nm}$  is a path length generated heuristically by the recent neighborhood.

#### IV. QOS ANYCAST ROUTING ALGORITHM BASED ON IMPROVED ANT COLONY OPTIMIZATION ALGORITHM

##### A. A Node Selection Strategy Embedded With Iterative Operator

For the reasons of positive feedback mechanism of ant colony algorithm, the more pheromone can be left on the optimum path. From (11) and (12), we know that the

more pheromone, the greater probability to be chosen the path in the next iteration. So, more and more ants choose the path with the increase of the iterations, even all the ants are do that.

However, once the path is not a global optimal path, only is a local optimal path, the entire ant colony would fell into a local optimum and the algorithm would stagnate prematurely [20-21].

The reason for above-mentioned situation is that the ant colony chooses the path with a higher fitness value by the positive feedback mechanism in the early exploration, the pheromone concentration of this path has been strengthened. However, the path that it is actually a better path is gradually “forgotten” due to the fewer ants on it in the initial phase. Consequently, the global search ability of ACO is weakened.

The optimizing process of ACO actually includes “exploration” and “utilization”. “Exploration” is to expand the search space of the ant colony and to get the global optimal solution, but this will reduce the convergence speed. “Utilization” aims mainly at historical experience of ant colony. The role of this process is in order to limit the search range nearby the higher fitness value. Nevertheless, this may cause the algorithm to stagnate precociously. So, how to find a fine balance between the “exploration” and “utilization” is important.

In order to find a balance between the precocious, stagnation and convergence speed, the node selection formula of the ant colony algorithm is redefined, which can be represented as follows:

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{i,j}^\alpha(t) \eta_{i,j}^\beta(t)}{\sum_{k \in allowed_k} (\tau_{i,j}^\alpha(t) \eta_{i,k}^\beta(t))} & \text{if } j \in allowed_k \text{ and } q > q_0(l); \\ \arg \max \{ \tau_{i,j}^\alpha(t) \eta_{i,j}^\beta(t) \} & \text{otherwise;} \end{cases} \quad (17)$$

$$q_0(l) = 1 + \frac{1}{2} (e^{l/\maxCycles} - e) \quad (18)$$

Where  $q$  is a random number in  $[0,1]$ ,  $\maxCycles$  is the maximum number of iterations,  $l$  is the iteration(iterative operator). It's easy to see,  $q_0(l)$  is an increasing function in  $(0,1]$ . so  $q_0(l)$  is smaller in the initial stage of the algorithm,  $q$  bigger than  $q_0(l)$  with greater probability. So,  $p_{ij}^k(t)$  equal to the upper part of (17) with greater probability as well, which show that it is fully reflects the randomness to calculate the weighted value of visibility and pheromone on the path of each optional path and to select path according to the probability, this mechanism conducive to expand the algorithm's global search. At the later phase of the iteration,  $q_0(l)$  is bigger,  $q$  smaller than  $q_0(l)$  with greater probability, so,  $p_{ij}^k(t)$  equal to the lower part of (17) with greater probability as well, which demonstrate that the ants to select which path is determined by the weighted size of visibility and

pheromone is maximum. Which reflects the certainty (randomness weaken) , and speeds up the convergence rate.

*B. Self-Adaptive Adjustment for the Volatile Factor*

In ACO, artificial ants have memory, with increasing of the iteration, the pheromone will gradually evaporate. In ACO, we use parameter  $\rho$  as is the evaporation rate (or pheromone volatility), and  $1-\rho$  is the rest on the path[22].

Ant colony optimization algorithm has several shortcomings, for example, slow convergence speed, easy to get into a local optimum, etc, as well as the evolutionary algorithms such as genetic algorithms.

The size of the evaporation rate  $\rho$  is directly correlative with the global search ability and convergence rate of ACO. Due to the existence of evaporation rate  $\rho$  , that the pheromone on the never searched path (or feasible solution) will reduce to close to 0 when the problem scale is larger, thus, the global search ability of ACO will be decreased. If the evaporation rate  $\rho$  is too large, the visited search path will be selected again with a larger possibility. Due to domination of positive feedback effect of pheromone, randomization will be diminished. Although the algorithm can converge faster, it is easy to fall into a local optimum. Conversely, if we decrease the evaporation rate  $\rho$  , the randomization of ACO and global search capabilities can be improved. The pheromone remain on the path will be dominant; the stochastic searching ability will be enhanced. However, the convergence rate will be slowed down[23-25].

Therefore, how to select the evaporation rate  $\rho$  of ant colony algorithm must be considered of two performance indicators: global search ability and convergence speed. In this article, the  $\rho$  can be changed self-adaptively according to the distribution of ants.

Definition 1. Let the number of path of m ants on original node s move to the objective node d at current iteration is r (including qualified path and failed path), the dispersed degree of ants at j-th iteration can be defined as follows:

$$dis(j) = \sqrt{\frac{r}{m}} \tag{19}$$

Obviously, the more the number of paths of ants searched during the j-th iteration, the dispersed degree  $dis(j)$  is greater. On the contrary,  $dis(j)$  is small, these ants will be concentrated.

Extreme cases: (1) When the m ants pass different paths, each path has different ant,  $dis(j) = 1$  , this demonstrate that the ants are exceedingly dispersed. (2)

The m ants walk on the same path,  $dis(j) = \sqrt{\frac{1}{m}}$  , which show that the ants are extremely concentrated.

Therefore, we can self-adaptively adjust the evaporation coefficient  $\rho$  according to the distribution of ants.

(1) In continuous Z times iterations, the dispersion degree  $dis(j) > W$  (W is a given constant), which indicate the distribution of ants is relatively dispersed. It may lead to slow convergence rate, we should strengthen the positive feedback and increase evaporation coefficient  $\rho$  . The aim is to accelerate convergence rate.

The  $\rho$  can be adjusted by (20):

$$\rho(t) = \begin{cases} \varphi \bullet \rho(t-1) & (\rho \leq \rho_{max}) \\ \rho_{max} & (\rho > \rho_{max}) \end{cases} \tag{20}$$

Here  $\rho_{max}$  is the maximum of  $\rho$  .

(2) In continuous Z times iterations, the dispersion degree  $dis(j) < Y$  (Y is a given constant), which indicate the distribution of ants is relatively concentrated, It may lead to prematurity or stagnation, should increase the diversification of path and reduce appropriately evaporation rate  $\rho$  . The aim is to enhance diversification and global search capability. The  $\rho$  can be updated by (21):

$$\rho(t) = \begin{cases} \lambda \bullet \rho(t-1) & (\rho \geq \rho_{min}) \\ \rho_{min} & (\rho < \rho_{min}) \end{cases} \tag{21}$$

Where  $\rho_{min}$  is the minimum of  $\rho$  ,

Thus, in each generation, the evaporation rate  $\rho$  can adjust self-adaptively according to the distribution of ants, which not only can ensure the convergence rate, but also can greatly enhance the global search ability of ant colonies.

*C. The Mutation And Secondary Ant Colony Operation*

For solving QoS anycast routing problem, these ants usually starting from the same source to find the destination node set. So, the ACO for solving QoS anycast routing is easier to occur premature convergence compared to solving TSP.

The judgment of premature convergence is the foundation of premature processing. Experiments show that both premature convergence and global convergence of the ant population, the evolutionary speed will be markedly slowed down and this evolutionary speed can be used as one of judgment condition of the premature convergence.

We let  $f_{best}(t)$  is the optimal value at t th generation.

$f_{best}(t-1)$  is the optimum value at t - 1 th iteration, the ants evolutionary speed  $\theta(x)$  can be expressed as follows:

$$\theta(x) = \frac{f_{best}(t)}{f_{best}(t-1)} \leq 1 \tag{22}$$

We can know that the global optimal value is decided by each individual's optimal value from the mathematical model of ACO, and current global optimal always equal

to or better than the optimal value of previous iteration. The  $\theta(x)$  take full account of the evolutionary speed of ant colony. In Generally, the change of  $\theta(x)$  is larger in the early iteration, the ant populations evolve faster, after empty iterations, the  $\theta(x) = 1$ , which indicates that the algorithm fall into the prematurity or find the optimal values.

If  $\theta(x)$  remains 1 in a given continuous C times iterations, we may consider the algorithm is getting into local optimization and should mutate or secondary colony operations for the optimal path, so that to jump out of a local optimum.

The mutation and secondary ant colony operation are specific described as follows: (1) a node is added that using as counter NodeAppSum (1,N), each node count will be automatically increased by 1 once a ant go through the node. (2) Let optimal path  $P_n = \{v_s, v_i \dots v_j, v_d\}$  at n-th iterations. If the number of nodes between  $v_i$  and  $v_j$  is greater than or equal to 1, we need perform mutation, otherwise, nothing need do. We assume that the path has R nodes from node  $v_i$  to node  $v_j$ , to select consecutive R/3 nodes  $\{v_m, v_o \dots v_n\}$  randomly, and to create an adjacency list for each node. For each node, the connected nodes with it are separately added in their adjacency list, and the neighboring node is selected that having minimal NodeAppSum (1,N) in the adjacency list. If the minimal NodeAppSum (1,N) have two or more nodes, we select randomly a node. At last, we replace the corresponding nodes  $\{v_m, v_o \dots v_n\}$  with the selected nodes  $\{v'_m, v'_o \dots v'_n\}$ , sequentially, we can get the mutated path  $\{v_s, v_i, v'_m, v'_o \dots v'_n, v_j, v_d\}$ .

Calculate the fitness of new path, if the fitness of new path is superior to the fitness of original path, we retain the better solution, otherwise, the mutation and secondary ant colony operation are performed again between  $v_m$  and  $v_n$  until the count equal to X.

Due to the mutation and secondary ant colony operation, the algorithm can successfully escape from local optima, and can rapidly approximate to the global optimum.

#### D. Describing Of Algorithm Step

**Step1:** initialize parameters. Set the number of ants is  $m$ , maximum iteration is Num, source node is s, the set of target nodes is M. define  $\alpha, \beta, Q, \phi, \lambda, C, Z, W, Y, X, \rho_{\max}, \rho_{\min}$  and these constraints D,DJ,B,PL. initialize pheromone  $\tau_0$  of each edge, evaporation rate  $\rho, \rho_0$ , let iteration  $l=0$ .

**Step 2:** simplify network. To get the new network topology by removing the links and nodes that violating

the constraint conditions, and to route based on the new network structure.

**Step 3:** update  $l = l + 1$ , for each link, to set the increment of the pheromone.  $\Delta\tau_{ij} = 0, t = 0, i=1$ , initialize the m ants' position on the source and generate a tabu list for each ant, meanwhile, the source is added into the tabu list.

**Step 4:** let  $t = t + 1$ , set the current target node is  $M_i$ .

For each ant  $k$  that they don't complete their search process, these ants will select to move to node  $N_j$  from node  $N_i$  by (17). Meanwhile, for the path from source node s to node  $N_j$ , the all constraint conditions will be estimated. If the node  $N_j$  does not exist, it means the ant is dead, and recorded the path that this ant passed. If  $N_j$  is the target node  $M_i$ , it indicate the search process of this ant is finished. Otherwise, make the  $N_j$  will be added into tabu list of ant k, and continue to search.

**Step 5:** Repeat step 4 until all ants have completed their search, and record the qualified path  $P(s, M_i)$  from source node to the target node  $M_i$ .

**Step 6:** Calculate the fitness of the qualified path  $P(s, M_i)$ , and compare the fitness with the fitness of current global optimal path, if the fitness of  $P(s, M_i)$  is better, we update the global optimal path  $Path_{best} = P(s, M_i)$ . Locally update the pheromone on the path  $P(s, M_i)$  by (15).

**Step 7:** Set  $i=i+1$ , continue to search qualified path from the source node to other destination nodes by repeat Step4, Step5 and Step 6.

**Step8:** The current global optimal path  $Path_{best}$  can be obtained after the each ant visited all target nodes in a generation.

**Step9:** Adjust self-adaptively pheromone evaporation rate  $\rho$  according to the above method, and update global pheromone of the links on the best path by (13)

**Step10:** If  $\theta(x)$  remains 1 in a continuous C (given constant) iterations, then to perform the mutation and secondary ant colony operation for the optimal path.

**Step11:** If  $l < \text{Num}$ , then perform Step 3, otherwise, execute Step 12.

**Step12:** Output the optimal path  $l < \text{Num}$ , terminate algorithm.

## V. SIMULATION

The proposed improved ACO is implemented in MATLAB, In order to validate the validity of improved ACO we selected an example to experimentize. As shown in Figure 1, the network includes 30 nodes. In our

experiment, we only consider the constraints: link bandwidth, delay and cost, so, the link attribute can be described with a 3-tuple(bandwidth, latency, cost). Nevertheless, computation of multiplicative property can reference additivity.

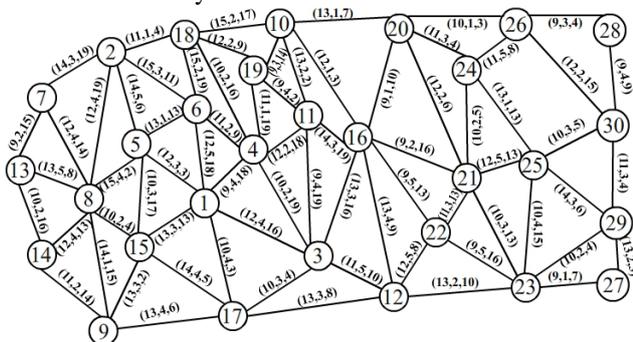


Fig 1. Network Model

Source node  $s = 30$ , target node is 2 and 9, bandwidth requirement  $B = 8$ , delay requirement  $D = 46$ . parameters setting as follows:  $\alpha = 1$ ,  $\beta = 2$ ,  $\rho_0 = 0.2$ ,  $\rho = 0.3$ ,  $\rho_{max} = 0.9$ ,  $\rho_{min} = 0.1$ ,  $Q = 2$ ,  $\lambda = 0.95$ ,  $C = Num/5$ ,  $Z = 3$ ,  $W = 0.85$ ,  $\phi = 1.05$ ,  $Y = 0.45$ ,  $X = 5$ . For each link, the initial pheromone is 1, the population size  $m = 10$ , these ants at the source node in the initial phase, and the number of iterations is 20.

Figure 2 is the evolutionary curve of cost and delay for ACO and improved ACO to search the optimal path in 20 iterations, where (Y-axis is cost and delay, the unit of cost is thousand yuan, the unit of delay is second, X-axis is iteration).

Table 1 is the comparison of experimental results that using ACO and improved ACO to solve the QoS anycast routing problem.

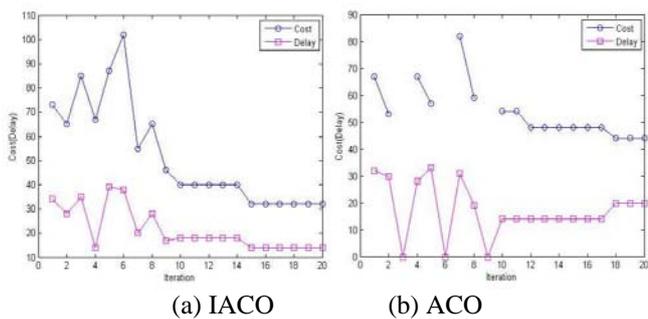


Fig 2. The evolutionary curve of cost and delay.

TABLE 1.

COMPARISON OF EXPERIMENTAL RESULTS

algorithm	Optimum path	Cost	Delay	Iteration
improved ACO	30-29-23-12-17-9	32	14	15
ACO	30-29-27-23-12-3-17-9	44	20	18

From Figure 2, we can see that there are some fluctuations of the cost and delay in evolutionary curve of improved ACO curve in the earlier-iteration, it

demonstrate that the global search ability of IACO is enhanced. From Table 1, we know that experimental results of IACO are more remarkable. For the same network model, and the same QoS anycast routing constraints condition is satisfied, the IACO to find the optimal path only expend 15 iterations. However, the basic ACO to find the optimal path need 16 iterations and the optimal path is a local optimum. The simulation results show that the IACO have preferable global search ability and can effectively jump out the local minima, consequently, the IACO can faster converge to the global optimal solution. So, the IACO is a feasible and effective algorithm.

VI. CONCLUSION

ACO has several shortcomings that the convergence rate is slow and it's easily to stuck in local optimum. Aiming at this shortcomings and balance problem between the "exploration" and "utilization". We proposed an improved ACO that combined with characteristic of QoS Anycast Routing problem. The iteration operator is embedded in the node selection strategy, and expands the scope of the global search; the evaporation rate can adjust self-adaptively according to the distribution of ants, which not only can ensure the convergence rate, but also can greatly enhance the global search ability of ant colonies. According to the evolutionary speed of the population, the premature convergence is estimated. The mutation and secondary ant colony operation is introduced, which can make the algorithm successfully to escape from local optima, and can rapidly approximate to the global optimum. The simulation results showed that the proposed algorithm is feasible and effective.

ACKNOWLEDGMENT

This work is supported by the Application Research Program (Natural Science Foundation) of Guangxi Province of China (No. 2010GXNSFB013052), and the Research Program (Youth Foundation) of Guagnxi University for Nationalities of China (No. 2011MDQN040).

REFERENCES

- [1] S. Weber, L. Chen. "A server of anycast in IPV6 networks," *IEEE Communications Magazine*, Vol. 42, pp.127-132, 2004.
- [2] K. J. Lee, W. H. Choi, J. S. Kim, "A study of anycast application for efficiency improvement of multicast trees", *Lecture Notes in Computer Science*, Vol. 3036, pp. 569-572, 2004.
- [3] C. P. Low, X. Song, "On Finding Feasible Solution fot the Delay Constrained group multicast problem", *IEEE Transaction on Computers*, Vol. 51, pp. 581-587, 2002.
- [4] W. Jia, D. Xuan, W. Zhao, "Integrated routing algorithm for anycast messages", *IEEE Communications Magazine*, Vol. 38, pp. 48-53, 2000.
- [5] T. S. Li, S. Q. Chen, M. Wang, "Application of an Improved Particle Swarm Optimization Algorithm in QoS Anycast Routing" *Jornal of Chinese Computer Systems*, Vol. 31, No. 1, pp. 67-71, 2010.

- [6] Y. S. Xiang, X. B. Yue, X. Y. Tang, "QoS Anycast Routing Based on Immune Genetic Algorithm" *Computer Engineering*, Vol. 34, No. 10, pp. 207-209, 223, 2008.
- [7] K. Li, S. Y. Fang, "Delay-constrained Anycast Routing Algorithm Based on Improved Genetic Algorithm", *Computer Engineering and Design*, Vol. 29, No. 15, pp. 3874-3877, 2008.
- [8] T. S. Li, Z. G. Zhao, "An Anycast Routing Algorithm Based on Simplified Particle Swarm Optimization and Diversity Strategy", *Proceedings of 6th International Conference on Computer Science & Education, SuperStar Virgo, Singapore*, pp. 1075-1079, 2011.
- [9] M. Dorigo, "Optimization, Learning and Natural Algorithms(in Italian)," *Ph.D. dissertation, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy*, 1992.
- [10] M. Dorigo, V. Maniezzo, A. Colomi, "Positive Feedback as a Search Strategy," *Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy*, Tech. Rep. 91-016, 1991.
- [11] M. Dorigo, V. Maniezzo, A. Colomi. "Ant system: Optimization by a colony of cooperating agents", *IEEE Trans on Systems, Man and Cybernetics.B*, Vol. 26, pp. 29-41, 1996.
- [12] J.Y.Li, S.D.Sun, Y.Huang. "Adaptive Hybrid Ant Colony Optimization for Solving Dual Resource Constrained Job Shop Scheduling Problem", *Journal of Software*, Vol.6, No.4, pp.584-594, 2011.
- [13] M. Xiao, T. S. Li, "Application of ant colony algorithm in delay constrained anycast routing", *Computer Engineering and Applications*, Vol. 44, No.24, pp. 121-123, 2008.
- [14] T. S. Li, S. Q. Chen, J. E. Chen, "Research on Adaptive Anycast Routing Algorithm with Multiple QoS Parameters", *Journal of Computer Research and Development*, Vol. S1, pp. 415-420, 2008.
- [15] M. Yang, T. S. Li, "A RDO-PSO Algorithm for Anycast Routing with Multi-QoS Constraints", *Proceedings of 4th International Conference on Computer Science & Education*, Nanning, China, pp. 376-379, 2009.
- [16] T. S. Li, M. Xiao, "An Improved Ant Colony Optimization Algorithm for Multiple QoS Anycast Routing", *Proceedings of 2010 International Conference on Computer and communication Technologies in Agriculture Engineering*, Chengdu, China, pp. 544-547, 2010.
- [17] M. Dorigo, L. M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," *IEEE Transactions on evolutionary computation*, Vol. 1, No. 1, pp. 53-66, 1997.
- [18] L. M. Gambardella, M. Dorigo, "Solving symmetric and asymmetric TSP's by ant colonies," *Proc.IEEE Int. Conf. Evolutionary Computation, IEEE-EC 96*, pp. 622-627, 1996.
- [19] N. Meuleau, M. Dorigo, "Ant colony optimization and stochastic gradient descent ," *Artif.Life*, Vol. 8, No.2, pp. 103-121, 2002.
- [20] L. Chen, J. Shen, L. Qin, "An adaptive ant colony algorithm based on equilibrium of distribution", *Journal of Software* , Vol.14, No.8, pp.1379-1387, 2003.
- [21] X. H. He, X. B. Hu, J. D. Zhao, "Ant Colony Optimization Algorithm Based on Adaptive Transition Probability", *Computer Engineering*, Vol.36, No.23, pp.165-167, 2010.
- [22] Z.B.Liu, L.Zhang, X.S.Meng. "A Novel Hybrid Stochastic Searching Algorithm Based on ACO and PSO: A Case Study of LDR Optimal Design", *Journal of Software*, Vol.6, pp.56-63, No.1, 2011.
- [23] S. C. Zhan, J. Xu, J. Wu, "The Optimal Selection on the Parameters of the Ant Colony Algorithm" , *Bulletin of science and technology*, Vol.19, No. 5, pp. 381-386, 2003.
- [24] J. Zhang, X. M. Hu, X. Y. Luo, *Ant Colony Optimization*, Beijing: Tsinghua University Press, 2006.
- [25] M. Birattari, P. Pellegrini, M. Dorigo, "On the Invariance of Ant Colony Optimization", *IEEE Transactions on Evolutionary Computation*, Vol. 11, No. 6. pp. 732-742, 2007.



**Yongsheng Li** born in Guangxi, China, 1979, has got a MS degree in Computer Application Technology from Guagnxi University for Nationalities in 2009. He is currently a associate professor at School of Information Science and Engineering, Guangxi University for Nationalities. His research work includes recognition, computational Intelligence and its application et al.