Front. Electr. Electron. Eng. China 2007, 2(3): 268–272
DOI 10.1007/s11460-007-0049-0

RESEARCH ARTICLE

ZHU Qingbao, WANG Lingling

The analysis of the convergence of ant colony optimization algorithm

Abstract The ant colony optimization algorithm has been widely studied and many important results have been obtained. Though this algorithm has been applied to many fields, the analysis about its convergence is much less, which will influence the improvement of this algorithm. Therefore, the convergence of this algorithm applied to the traveling salesman problem (TSP) was analyzed in detail. The conclusion that this algorithm will definitely converge to the optimal solution under the condition of $0 < q_0 < 1$ was proved true. In addition, the influence on its convergence caused by the properties of the closed path, heuristic functions, the pheromone and q_0 was analyzed. Based on the above-mentioned, some conclusions about how to improve the speed of its convergence are obtained.

Keywords ant colony optimization algorithm, convergence analysis, heuristic function, TSP

1 Introduction

Originally proposed by Dorigo et al., the ant colony optimization (ACO) [1] has been applied widely in many fields. Although this algorithm has been one of the focuses of the studies, the ACO algorithm is a new bionic approach, whose theoretical study attracted less attention. In most of the researches, only algorithms and application methods are shown, whereas the analysis about the convergence is not given. Until recently, some scholars have given the convergence proof of certain algorithms [2–5], but the relevant researches and papers are still much fewer. The serious lack of theories has become the bottle-neck in the improvement and development of the algorithms. The convergence property of the algorithm is proved in previous papers [2–5], which analyzed the convergence property of different methods. For example, in Ref. [5], the basic idea is to utilize random processes, which is used to prove the convergence of the algorithm, to develop a branching ant algorithm according to the number of ants, paths, etc. In Refs. [3] and [4] the convergence proof of the graph-based ant algorithm is given, whose foundation lies in the pheromone intensity on the optimal routes, which is described as a non-homogeneous Markov random process based on discrete time. In this algorithm, there is a single optimal solution to which only the global pheromone can be updated, and there is no updating of local pheromone and all ants are first positioned at the same node. In Ref. [2], the convergent property is analyzed through the pheromone. In this paper we take the convergence property of this algorithm applied to TSP for example, and consider sufficiently the influence on its convergence caused by heuristic functions, the pheromone, q_0 and the properties of the path, etc. Then we produce a new analysis of the ant colony optimal algorithms applied to TSP. Based on this, some improvement measures are suggested in the conclusion.

2 Description of problem and definition

For the purpose of discussion without loss of generality, we apply the ACO algorithm to TSP for example. We have the following definitions for the convenience of illustration.

Let AS be a finite convexed polygon region in which n cities are distributed. $C = \{c_1, c_2, \ldots, c_n\}$ is a set of these cities where $R = \{1, 2, \ldots, n\}$ is the city sequence number. Assume we have a Cartesian coordinate $\sum_0$ in AS. $\forall c_i \in C$, $i \in R$ has a definite coordinate (x_i, y_i) in $\sum_0$, which is marked as $c_i(x_i, y_i)$. The link-line between any two cities forms an edge which is marked as e_{ij}, $i, j \in R$.

Definition 1 Let $d(c_i, c_j)$ (simply d_{ij}) be the distance or side length between any two cities c_i and c_j determined by Eq. (1), simply d_{ij} and we have $d_{ij} = d_{ji}$, $i, j \in R$

$$d(c_i, c_j) = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \tag{1}$$

Translated from *Control and Decision*, 2006, 21(7): 763–766 [译自: 控制与决策]

ZHU Qingbao (✉), WANG Lingling
Department of Computer Science, Nanjing Normal University, Nanjing 210097, China
E-mail: Zhuqingbao@njnu.edu.cn

Definition 2 Let ant = $\{1, 2,\ldots, k,\ldots, m\}$ be a set of m ants, $\forall k \in$ ant is an ant, m is the total number of ants in the set, then $\tau_{ij}(t)$ is the amount of pheromone information on a route $e_{ij}(i, j \in R)$ due to some ants at t instant.

Definition 3 Let ant k be at location P in AS at a random time instant. $\forall P$ has a definite coordinate (x, y) in $\sum_0$, we have $P(x_i(t_i), y_i(t_i))$ (simply P_i or $P(t_i)$). If the coordinate of P_i is equal to $c_j(x_j, y_j) \in \mathrm{AS}$, the P_i is equivalent to c_j, which is marked as $P_i - c_j$. Let $tabu_k$ be the node set where the ant has already travelled. Obviously $tabu_k$ is a dynamic set including the nodes that ant k has already covered. If a node has already been in the set, it cannot be chosen by ant k in the future, $tabu_k$ is hence referred to as forbidden table, and $V = C\text{-}tabu_k$ is the node set from which the future destination node can be chosen.

Definition 4 Let $T = \{t_0, t_1,\ldots, t_i,\ldots, t_{n+1}\}$ $t_0 < t_1 < \ldots < t_i < \ldots < t_{n+1}$, and t_{n+1} be a finite time instant, $\forall P \in \mathrm{AS}$, $\forall p(t_i) \in \mathrm{FS}$, $i \in R$, the continuous mapping f: $T \rightarrow \mathrm{AS}$ makes $f(t_0) = P_0$, $f(t_1) = P_1,\ldots, f(t_i) = P_i,\ldots, f(t_n) = P_0$, $i, j \in R$. We refer to f as the map of a close route starting from P_0 and returning back to P_0. The mapping set $f(T)$ is referred to as a close path from P_0 to P_e, which is also a continuous curve connecting node P_0 and P_e in *AS*. The length of this route is marked by $L(P_0, P_e)$ or simply L calculated by Eq. (2), where d_l is from Eq. (1)

$$L = \sum_{l=1}^{e+1} d_l;\ d_l = d(c_i, c_j),\ c_i, c_j \in C,\ i, j \in R \tag{2}$$

Definition 5 Let $\mathrm{BR}_i(c_i(x_i, y_i)) = \{c \mid c \in C, d(c, c_i) \leqslant \mathrm{d}_{\min}\}$ be the neighbouring node set of node c_i, where $d_{\min}$ is the distance threshold determined by the concrete problem and required size of neighbouring node set. If z represents the travelable node set in the neighbouring domain of node c_i, we have $z = (C - tabu_k) \cap \mathrm{BR}_i$.

Definition 6 $\eta_{ij} = 1/d(c_i, c_j)$ is the heuristic function, which causes an ant to choose the next node j as it is at node i.

3 ACO algorithm

In order to prove conveniently, according to the above description and definition integrated with the ACS algorithm, the steps of the ant algorithm are described as follows.

Step 1 Initialization. Assign m ants to n cities randomly, and these cities are included in the forbidden table $tabu_k$. Also set a numeric counter NC = MAX, and preset the values of β,α and ρ, let $\forall \tau_{ij} = \tau_0;\ i, j \in R$.

Step 2 $\forall k$, take the current city i as the center, and select the next city $j \in R$. According to Definition 3, select $|V|$ virgin cities $\{j_1, j_2,\ldots, j_p, j_v\}$ from $n-1$ cities, i.e. $j_p \notin tabu_k$.

Step 3 Select the next city j from the candidate city. Set V according to Eqs. (3) or (4)

$$j = \begin{cases} \arg\max\limits_{j \notin tabu_k} \{\tau_{ij}(t)][\eta_{ij}(t)]^{\beta}\}; & if \quad q \leqslant q_0 \\ S; & \text{otherwise} \end{cases} \tag{3}$$

In this equation, $0 < q_0 \leqslant 1$ is the initial parameter, $q \in (0, 1)$ is a random number, and S is a stochastic variable defined by Eq. (4)

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)][\eta_{ij}]^{\beta}}{\sum\limits_{h \notin tabu_k} [\tau_{ih}(t)][\eta_{ih}]^{\beta}}; & j \notin tabu_k \\ 0; & j \in tabu_k \end{cases} \tag{4}$$

In this equation, $p_{ij}^{k}(t)$ represents the transferring probability that ant k move from node i to node j and β is the degree of importance of the heuristic information. If $q > q_0$, then the transferring probabilities p_{ij}^{k} of $|V|$ cities are computed and the next city j is selected according to the roulette wheel rule.

Add j to the forbidden table $tabu_k$.

Step 4 Local pheromone update.

We use parameter $1-\rho$ to denote the degree of evaporation of the pheromone. As an ant passes an edge between two nodes, the local pheromone trail on that edge will be updated according to Eq. (5).

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \rho\Delta\tau_{ij};\quad \Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{5}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{l_{jb}}; & \text{when ant } k \text{ passes edge}(i, j) \\ 0; & \text{otherwise} \end{cases}$$

$$\text{if } \tau_{ij}(t+1) < \tau_{\min},\ \text{then } \tau_{ij}(t+1) = \tau_{\min}$$

Here, Q is a constant, l_{jb} is the distance k has passed during this time, $\Delta\tau_{ij}$ is the quantity of pheromone trail released by ant k on edge (i, j) in this circle, and $\Delta\tau_{ij}$ is the pheromone increment on edge (i, j) in this cycle. $\tau_{\min}$ is the least pheromone intensity of limits, which is a small constant as well.

Step 5 After m ants have selected their nodes, make the newly selected nodes as the starting nodes and the algorithm returns back to Step 2 for the next nodes till all ants have passed all the nodes.

Step 6 After m ants have passed all the nodes, the route length L_k passed by ant k is calculated with Eq. (2). The minimal value of L_k, which is marked as $L_{k\min} = \min L_k$, is further found and kept in memory.

Step 7 Global pheromone update.

After all ants have passed all the nodes, the global pheromone trail on the path will be updated according to Eq. (6)

$$\tau_{ij}^{\text{new}} = (1-\alpha)\tau_{ij}^{\text{old}} + \alpha\Delta\tau_{ij} \tag{6}$$

$$\Delta\tau_{ij} = \begin{cases} \dfrac{1}{l_k}; & \text{if } e_{ij} \in \text{global-best-tour} \\ 0; & \text{otherwise} \end{cases}$$

$$\text{when } \tau_{ij}^{\text{new}} < \tau_{\min} \text{ setting } \tau_{ij}^{\text{new}} = \tau_{\min}$$

where α is the volatility coefficient of the global pheromone; l_k the distance of optimal path in this circle; $e_{ij}\in$global-best-tour means that the side e_{ij} ant k has passed belongs to the optimal path.

Step 8 Compare $L_{k\min}$ of the current iteration with the optimal route length l_d which has already been obtained, l_d is replaced by $L_{k\min}$ if $l_{k\min} < l_d$ and meanwhile the optimal route table will be updated.

Step 9 If the counter value minus 1 is not equal to zero, then reset and initialize the tabu table and repeat the whole process until the counter is zero.

4 Analysis of convergence

Definition 7 Assume that there is a closed route f_i, $\forall t_1$, $t_2\in[t_0,\ t_n]$, where $t_2 = t_1 + \Delta t$, Δt is the time ant k needs to travel from the current node to the next one. If we always have $d(P(t_1)\ ,\ P(t_2)) = \min\ \{d(P(t_1),P),P\in V\}$ and $L(P(t_2),\ P_e) < L(P(t_1)\ ,\ P_e)$, then $f_i\ (t_2) = P(t_2)$ is a node which monotonically approximates node P_e, and f_i is a closed route, which monotonically approximates P_e from P_0. We include all the monotonically approximated routes in a set marked as F_p, and all the non-monotonically approximated routes, f_n, in another set marked as F_N.

Lemma 1 $\forall e_{ij}$, we have $\tau_{\min}\leqslant\tau_{ij}\leqslant\tau_{\max}$, where $\tau_{\max} = \max\left(\frac{\tau_m}{\rho},\frac{\tau_m}{\rho}\left[\frac{(1-\alpha)\rho+\rho}{(1-\alpha)\rho+\alpha}\right]\right)$, $\tau_{\mathrm{m}} = \max\ (\rho\Delta\tau_{ij},\alpha\Delta\tau_{ij})$

Proof Set the optimal path of some generation as f_0. According to arithmetic steps and Eqs. (3)–(6), the following three cases can be classified as follows.

1) The side e_{ij} has not been passed by ant k and the pheromone on it only clears off without increasing. According to Eq. (5), there is

$$\tau_{ij}^{\min}(t) = (1-\rho)^t\tau_0,\ t\to\infty,\ \tau_{ij}\to 0,\ \tau_{ij} = \tau_{\min}\ \text{(confined to } \tau_{\min}\text{)} \tag{7}$$

2) k moves across $e_{ij}\in f_i$ *and* $e_{ij}\notin f_0$, so the pheromone on it is locally updated. According to Eq. (5) there is

$$\tau(1) = (1-\rho)\tau(0) + \tau_m;\ \text{where } \tau_m = \rho\Delta\tau_{ij}$$

$$\tau(2) = (1-\rho)\tau(1) + \tau_m = (1-\rho)^2\tau(0) + (1-\rho)\tau_m + \tau_m$$

$$\vdots$$

$$\begin{aligned}\tau(t) &= (1-\rho)^t\tau(0) + (1-\rho)^{t-1}\tau_m + (1-\rho)^{t-2}\tau_m \cdots + \tau_m\\ &= (1-\rho)^t\tau(0) + \sum_{i=1}^{t}(1-\rho)^{t-i}\tau_m\end{aligned}$$

$$\tau_{ij}^{\max}(t) = (1-\rho)^t\tau_0 + \sum_{i=1}^{t}(1-\rho)^{t-i}\tau_m;\ \tau_{\mathrm{m}} = \rho\Delta\tau_{ij},$$

$$t\to\infty, \tau_{ij}^{\max}(t)\to\frac{1}{\rho}\tau_m$$

where $\sum_{i=1}^{t}(1-\rho)^{t-i}$ is a geometric series with ratio 1 minus ρ, and the sum of their first t items is $1-(1-\rho)^t)/(1-(1-\rho))$ so we get

$$\tau_{ij}^{\max}(t) = (1-\rho)^t\tau_0 + \sum_{i=1}^{t}(1-\rho)^{t-i}\tau_m,$$

$$\tau_m = \rho\Delta\tau_{ij}, t\to\infty,\ \tau_{ij}^{\max}(t)\to\frac{1}{\rho}\tau_m \tag{8}$$

where $\tau(0)$,$\tau(1)$,..., $\tau(t)$ represent the pheromone intensity of the initial state after the first generation of search till t generation of search respectively.

3) k moves across e_{ij} and $e_{ij}\in f_0$, then the pheromone on this side will be updated locally and globally. According to Eqs. (5)–(6) the following can be obtained

$$\begin{aligned}\tau_{ij}^{\max}(t) = {} & (1-\rho)^t(1-\alpha)^t\tau_0 +\\ & \sum_{i=1}^{t}(1-\rho)^{t-i}(1-\alpha)^{t-i}\left[(1-\alpha)\tau_m + \tau_m\right];\\ & t\to\infty,\ \tau_{ij}^{\max}\to\frac{\tau_m}{\rho}\left[\frac{(1-\alpha)\rho+\rho}{(1-\alpha)\rho+\alpha}\right]\end{aligned} \tag{9}$$

thereinto $\tau_m = \max\left(\rho\Delta\tau_{ij},\alpha\Delta\tau_{ij}^k\right)$

Comparing the above three cases, as to all $\forall\tau_{ij}$ all have $\tau_{\min}\leqslant\tau_{ij}\leqslant\tau_{\max}$. (End of proof).

Lemma 2 when $q_0 = 1,\forall e_{ij}\in f_i, f_i\in F_p,\forall e_{kl}\in f_n, f_n\in F_N$, we have $\tau_{kl}\leqslant\tau_{ij}$.

Proof When $q_0 = 1$, ant k will always use Eq. (3) to choose its next node. We get the following cases according to the algorithm process and Eqs. (3), (5)–(6).

1) $\forall e_{kl}(k,l\in R)\notin f_i$, $e_{kl}\in f_n$, in the travelable edge set, $\exists e_{ij}(i,j\in R)\in f$ which has $\eta_{ij} < \eta_{kl}$. Since in the initialization $\forall\tau_{ij}(i,j\in R) = \tau_0$, thus the probability that ant k selects and passes the edge e_{kl} using Eq. (3) is zero, therefore $\tau_{kl} = \tau_{\min}$.

2) $\forall e_{ij}\in f_i$, $e_{kl}\notin f_0$, $\tau_{ij}^{\max}$ is calculated by Eq. (8).

3) $\forall e_{ij}\in f_i$, $e_{kl}\in f_0$, τ_{ij}^{m} is calculated by Eq. (9).

We always have $\tau_{ij}\geqslant\tau_{kl} = \tau_{\min}$ in all the above three cases.

Theorem 1 If there exists a global optimal solution $f_0\in f_i$, when $q_0 = 1$, the algorithm can always converge to the optimal solution.

Proof When $\forall\ t_i\in[t_0,\ t_n]$, let the current position of ant k be $P(t_i)$, $P(t_i)$–c_i, from c_i, k chooses and proceeds to the next node c_j, c_j –$P(t_{i+1})$. Because the global optimal solution is $f_0\in f_i$, according to the Definitions (4), (7), there must exist a monotonically approximate node $P(t_{i+1})$, that is, $\forall P(t_{i+1})\in V$, $\exists P(t_{i+1})$ which makes

$$d(P(t_i),P(t_{i+1})) = \min\ \{d(P(t_i),P)P\in V\}$$

and satisfies that $L(P(t_{i+1}),P_e)$ is minimum, marked as $P(t_j)$. Suppose the probability that ant k chooses $P(t_j)$ from $P(t_i)$ is p, because $q_0 = 1$, then the algorithm will always choose c_j with Eq. (3). From Eq. (3) and Lemma 2 we can have $p\equiv 1$. Thus we can conclude that in t times node selection the probability of getting a global optimal solution is $p_l = 1-(1-p^n)^t \equiv 1$.

From the proof of Theorem 1 it is shown that the algorithm can rapidly converge to the optimal solution if the global optimal solution is f_i-type.

Proposition 1 If there exists an objective global optimal solution $f_0 \in f_i$, the convergence time on the condition of $q_0 < 1$ must be longer than $q_0 = 1$.

This proposition is quite obvious, which can be proved with the method of Theorem 2. The proof is omitted here.

Proposition 2 If there exists an objective global optimal solution $f_0 \notin f_i$, when $q_0 = 1$, the algorithm cannot converge to the global optimal solution.

Proof Since there exists an objective global optimal solution $f_0 \notin f_i$, $\exists e_{kl} \in f_n$, according to Lemma 2, when $q_0 = 1$, the probability of choosing a non-approximate node with Eq. (3) is zero. Suppose f_0 comprises x approximate nodes and y non-approximate nodes. Then suppose that the probability of choosing an approximate or a non-approximate node is R and r respectively, and thus the probability that k selects f_0 at any generation is $p = R^x r^y = 0$. (End of proof).

Theorem 2 If there exists an objective global optimal solution $f_0 \notin f_i$, when $0 < q_0 < 1$, the algorithm can always converge to the global optimal solution after a sufficiently long time.

Proof Suppose that a non-optimal F_{p_i} comprises x approximate nodes and y non-approximate nodes. Then suppose that the probability of choosing an approximate or a non-approximate node is R and r respectively, then R and r can be estimated as follows.

According to the algorithm, when $q \leqslant q_0$, k will select nodes based on Eq. (3). Otherwise, it will rely on Eq. (4). These nodes are either approximate or non-approximate. Setting p_3^x and p_3^y as the probability of selecting an approximate and a non-approximate node when $q \leqslant q_0$, p_4^x and p_4^y are respectively the probability when $q > q_0$, according to the algorithm

$$R = q_0 p_3^x + (1-q_0) p_4^x;\ r = q_0 p_3^y + (1-q_0) p_4^y$$

Based on Lemmas 1, 2, Definition 6 and Eq. (3), $p_3^y = 0$, $p_3^x = 1$, $\forall t > \tau_{\min} > 0$, $\eta_{\min} > 0$, therefore, the minimal limiting probability of choosing any nodes on path F_p by Eq. (4) is

$$p_{\min} \geqslant \frac{\tau_{\min}\eta_{\min}^{\beta}}{\sum\limits_{h\in(|v|-1)} \tau_{\max}\eta_{\max}^{\beta} + \tau_{\min}\eta_{\min}^{\beta}} > 0$$

Therefore, when $q > q_0$, the probability calculated by Eq. (4) is $p_{ij} > 0$, then $p_4^x > 0$, $p_4^y > 0$, while $0 < q_0 < 1$, hence we get $R > 0$, $r > 0$. The probability that k selects the optimal solution at least once after generation t is

$$p(t) \geqslant 1-(1-R^x r^y)^t;\ \text{when } t\to\infty,\ P(t)\to 1 \tag{10}$$

Namely, when $t = t_m$ is a relatively big value, we get

$$p(t) = 1-\varepsilon \approx 1 \tag{11}$$

(End of proof).

This theorem guarantees that, under the condition of $0 < q_0 < 1$, the algorithm can always converge to a global optimal solution after a sufficiently long time no matter how the pheromone changes.

It is also obvious that the convergence time will be quite long if some nodes of the optimal solution are not in f_i and the effect of pheromone is not considered.

The most serious effect of pheromone variation on convergence speed when $q_0 < 1$ will be discussed in the following text. Setting g as the ratio of the maximum and minimum pheromone, according to Lemma l, we get

$$g = \frac{\tau_{\max}}{\tau_{\min}} = \max\left(\tau_m/\rho, \frac{\tau_m[(1-\alpha)\rho+\rho]}{\rho[(1-\alpha)\rho+\alpha]}\right)/\tau_{\min}$$

Let the maximum probability of k choosing $P(t_{i+1}) \in f_0$ from $P(t_i)$ be $p_{\max}$ and $p_{\max}^{\tau}$, where the former is the one without considering pheromone, then we get

$$p_{\max} \leqslant \frac{{\eta_{\max}}^{\beta}}{\sum\limits_{h\in(|V|-1)} \overline{\eta}^{\beta} + \eta_{\max}^{\beta}} = \frac{b\overline{\eta}^{\beta}}{\sum\limits_{h\in(|V|-1)} \overline{\eta}^{\beta} + b\overline{\eta}^{\beta}} = \frac{b}{|V|-1+b} \tag{12}$$

here, $\overline{\eta}$ is the average of the heuristic function, $b\overline{\eta}^{\beta} = \eta_{\max}^{\beta}$. Then $p_{\max}^{\tau}$, the maximum probability of selecting an optimal node when considering pheromone will be estimated, setting $\forall e_{ij} \in f_0$, all have $\tau_{ij} = \tau_{\max}$, $\forall e_{ij} \notin f_0$, all have $\tau_{ij} = \tau_{\min}$, according to Eq. (4) we get

$$\begin{aligned} p_{\max}^{\tau} &\leqslant \frac{\tau_{\max}[\eta_{\max}]^{\beta}}{\sum\limits_{h\in(|V|-1)} \tau_{\min}\overline{\eta}^{\beta} + \tau_{\max}[\eta_{\max}]^{\beta}} \\ &= \frac{\tau_{\max} b\overline{\eta}^{\beta}}{\sum\limits_{h\in(|V|-1)} (\tau_{\max}/g)\overline{\eta}^{\beta} + \tau_{\max} b\overline{\eta}^{\beta}} = \frac{gb}{|V|-1+gb} \end{aligned} \tag{13}$$

Actually, the pheromone on the sides of the non-optimization solution is not all $\tau_{\min}$, hence, the probability is the ultimate maximum. The ratio of the two probabilities is

$$\frac{p_{\max}^{\tau}}{p_{\max}} = \frac{|V|-1+b}{\dfrac{|V|-1}{g}+b} > 1 \tag{14}$$

Generally, $g \gg 1$, and $|V|-1 \gg b$ in most cases, hence, we get

$$(p_{\max}^{\tau}/p_{\max}) \gg 1.$$

From Eqs. (10), (13)–(14), with the effect of pheromone, the convergence speed can be largely improved.

Proposition 3 The convergent speed must be improved by adopting the nearest neighbour strategy.

Proof Each time ant k only needs to choose nodes in z which is the travelable node set in the nearest neighbour node set, and then let $|z|$ replace $|V|$ in Eqs. (12)–(13). Since $|z| \ll |V|$, from Eqs. (10) and (13) the convergent speed is greatly increased. From the viewpoint of the algorithm, the employment of the nearest neighbour strategy can reduce the load of computation, which also leads to the speed improvement. (End of proof).

Proposition 4 The convergent speed will improve if the pheromone on the optimal routes are increased in a suitable range.

Proof According to Eqs. (13)–(14), in a suitable range a bigger g will lead to a bigger probability of choosing an optimal neighbouring node, hence from Eq. (10) the convergence speed will be increased. (End of proof).

The simulation experiments verified the analytical results in the present section, and the experimental results match well with these analytical results. The readers can see the relevant simulation experiments in Ref. [6].

5 Conclusions

In this paper, we analyze the convergence of the ant colony optimization algorithm applied to TSP. The conclusion is that, if the global optimal solution is a strictly approximate closed route, when $q_0 = 1$, the algorithm will surely converge to the optimal solution; when $q_0 < 1$ the convergence will slow down. If the global optimal solution is not a strictly approximate closed route, when $q_0 = 1$, the algorithm cannot converge to the optimal solution; when $0 < q_0 < 1$ it will take a much longer time for the algorithm to converge to the optimal solution. However, the convergence process will be accelerated by adding more pheromone to the optimal routes. Based on the conclusion, in the design, we should estimate or analyze the property of the optimal solution, and reasonably set q_0, then increase the pheromone on the optimal routes in a suitable range to improve the convergence speed.

To sum up, it is revealed, through the convergence analysis, that the two major factors that influence the convergence speed of ACO are pheromone and heuristic function. A suitable heuristic function is crucial for accelerating the convergence. If the value of the heuristic function is too large, it will restrict the effect of pheromone while conversely it will lead to a slow convergence. Moreover, the convergence speed can be significantly improved by adopting the nearest neighbour strategy, which is demonstrated by the convergence analysis.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant No. 60673102) and the Natural Science Foundation of Jiangsu Province of China (Grant No. BK2006218).

References

1. Colorni A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies. In: Varela F, Bourgine P, eds. Proceedings of ECAL91 European Conference of Artificial Life. Paris: Elsevier Publishing, 1991, 134–144
2. Thomas S, Dorigo M. A short Convergence proof for a class of ant colony optimization algorithms. IEEE Transactions on Evolutionary Computation, 2002, 6(4): 358–365
3. Amr B, Ahmed F. A proof of convergence for ant algorithms. Information Sciences, 2004, (160): 267–279
4. Walter J G. A graph-based ant system and its convergence. Future Generation Computer Systems, 2000, (16): 873–888
5. Walter J G. ACO algorithms with guaranteed convergence to the optimal solution. Information Processing Letters, 2002, (82): 145–153
6. Zhu Qingbao,Yang Zhijun. Ant colony optimization algorithm based on mutation and dynamic pheromone updating. Journal of Software, 2004, 15(2): 185–192 (in Chinese)