Computer-Aided Civil and Infrastructure Engineering **22** (2007) 44–55

A Parallel Ant Colony Algorithm for Bus Network Optimization

Zhongzhen Yang & Bin Yu

Transportation College, Dalian Maritime University, Dalian, 116026, China

&

Chuntian Cheng

Civil Engineering Department, Dalian University of Technology, China

Abstract: *This study presents an optimization model for a bus network design based on the coarse-grain parallel ant colony algorithm (CPACA). It aims to maximize the number of direct travelers per unit length, that is, direct traveler density, subject to route length and nonlinear rate constraints (ratio of the length of a route to the shortest road distance between the origin and destination). CPACA is a new optimal algorithm that (1) develops a new strategy to update the increased pheromone, called Ant-Weight, by which the path-searching activities of ants are adjusted based on the objective function, and (2) uses parallelization strategies of an ant colony algorithm (ACA) to improve the calculation time and the quality of the optimization. Data collected in Dalian City, China, are used to test the model and the algorithm. The results show that the optimized bus network has significantly reduced transfers and travel time. They also reveal that the proposed CPACA is effective and efficient compared to some existing ant algorithms.*

1 INTRODUCTION

In recent years, many studies have been done to improve the quality of bus networks. For example, Dubois et al. (1979) divided a bus network design into three steps: (1) identifying the roads needed for bus routes, (2) laying the routes, and (3) optimizing the operation frequencies. Hasselström (1981) proposed a two-stage model, which can simultaneously optimize routes and service frequencies. Baaj and Mahmassani (1995) argued that a bus network can be generated by optimizing the route and the frequency, simultaneously. Guan et al. (2004) used integer programming to optimize the line layout and assign trips in Minimum Spanning Trees. Sonntag (1979) approached the route plan problem in railways by a heuristic elimination method without appropriate simultaneous consideration of passenger route assignments. Wang and Yang (2001) and Simonis (1981) took a different approach from Sonntag (1979) by starting with an empty network, and then the routes with the most direct travelers on their shortest paths are successively added.

It has been found that the existing studies mainly contain two kinds of models. One combines the network design together with the service frequency, and the other solely concentrates on network design. Both of them are complicated in structure and need simplifications for solving. Especially, it becomes even more complicated when considering transfers. In addition, many algorithms attempting to solve the problems have been designed. For example, Steenbrink (1974) proposed a traditional mathematical programming method;

*To whom correspondence should be addressed. E-mail: *yangzhongzhen@263.net*.

Baaj and Mahmassani (1995) proposed a hybrid algorithm, which combines the path-searching algorithm with artificial intelligence and the bus system analysis in operational research. Dubois et al. (1979) identified a route layout by a heuristic algorithm; Chakroborty et al. (1995) solved the scheduling problem with an inherit genetic algorithm. All of them have applied implicit enumeration methods to solve the problems. It is known that those algorithms can be divided into five categories: (1) analytic method, (2) heuristic algorithm, (3) hybrid algorithm, (4) experience-based algorithm (Dashora and Dhingra, 1998; Fernandez, 1993), and (5) simulation model (Senevirante, 1990).

In this article, an optimization model is developed to maximize the direct traveler density based on demand for an entire bus network. At first, an empty network is built, and then routes are added in so as to maximize the direct traveler density until all passengers are loaded to the network or some constraints are overrun. Most existing studies (Simonis, 1981; Michael et al., 1997; Wang and Yang, 2001) firstly identified the shortest path between the origin and the destination and then sought the route with the most direct travelers. However, our method is not limited to the shortest paths, but seeks the path with the maximized direct traveler density from all possible routes. This is because travelers on the shortest path are not always the largest in number. Moreover, because the longer the route the more the travelers, some short paths may be abandoned if we try to maximize the number of direct travelers although they are rich in travelers. In addition, the overall length of an entire network increases when longer routes are laid, which consequently increases the operating costs without full utilization of the network or fleet. Therefore, to enhance network efficiency, we use the maximum direct traveler density as the objective function.

Network design is an NP-hard problem (Chakroborty, 2003; Garey and Johnson, 1979), which is difficult to solve via traditional methods. Recently many studies have proved that a Heuristic Algorithm is suitable for large-scale optimization problems. Here we use ACA in our model, which is an algorithm proposed by Dorigo (1992). It is a colony-based optimization approach, inspired by the food seeking actions of ant colonies. The algorithm does not focus on mathematical descriptions of specific problems, but rather on overall optimization ability and parallel capacity. Meanwhile, it benefits from better performance than earlier evolution algorithms such as the genetic algorithm and the simulated annealing one (Su and Weng, 2003; Dorigo et al., 1996; Yin, 2003). It has been successfully applied to some classic compounding optimization problems, for example, the traveling salesman problem (TSP), the quadratic assignment problem (QAP), and the job-shop scheduling problem (JSP). To make our model more effective, a new strategy for updating the increased pheromone is proposed, which considers both global and local information and allocates the increment of pheromone to links according to the traveler density of a route (global) and the link contribution to a route (local). The strategy is called Ant-Weight here. To expedite the calculations, parallel ACA based on coarse-grain strategy is carried out using a cluster of computers.

2 OPTIMIZATION MODEL

2.1 Urban Bus Network Design (*UBND*)

Urban bus network design should be based on passenger O-D matrix, and aim to facilitate trips as well as foster bus companies' profits (Chakroborty, 2003). The principles of the *UBND* are:

- Sections with the most travel demand should be fulfilled first and should be arranged for as much direct service as possible.
- The nonlinear rate (ratio of the length of a route to the shortest road distance between the origin and destination) should be as low as possible. A small nonlinear rate leads to a shortest travel distance/time.
- The network should be well accessible. The service area should extend to a wide area to reduce the unserved zones and ensure that the bus service is available within a short walking distance.
- The length of a route should be within a limit. Too long or too short routes should be avoided.
- The average number of travelers of a route should be over a minimum volume. Furthermore, priority should be given to the routes with high demand.
- The travelers of different routes should be balanced rationally.

2.2 Bus Network Optimization Model

Among existing studies, the direct traveler approaches (Wang and Yang, 2001; Simonis, 1981) are practical and convenient. They aim to maximize the number of the direct travelers under the length constraint or set the route along the shortest path. Maximization of the number of direct travelers may cause the bus routes in the network to be extremely long, whereas the latter limits the number of alternative routes and then damages the optimizing quality. Moreover, in both cases above, the route directions might not coincide with passenger flow directions. To overcome these disadvantages, we choose the maximum *direct traveler density* as the optimal objective, that is, the maximum number of travelers per unit length. As a bus route is composed of the stops and the road segments it covers, the optimization problem is hereby equivalent to finding the corresponding stops and road segments. Here, $\boldsymbol{N}$ denotes the stop collection; $\boldsymbol{A}$ denotes the collection of road segments. All of the alternative bus routes, the subcollections of $\boldsymbol{A}$, are denoted as $\boldsymbol{S} = \{S_{OD}\}$. Then, if not considering the loop route, the bus network optimization model is as follows:

$$\max D_{OD} = \frac{\sum_{i \in N} \sum_{j \in N} SP_{ij} x_{ij}}{\sum_{i \in N} \sum_{j \in N} \Delta_{ij} l_{ij} x_{ij}}$$

$$\text{s.t.} \quad \begin{cases} L_{\min} \le L \le L_{\max} & (a) \\ Q^{\text{sum}} > Q_{\min} & (b) \\ q^x \le q^x_{\max} & (c) \\ b^n \le b^n_{\max} & (d) \\ Q^{ij} < Q^{ij}_{\max} & (e) \\ \forall l_{ij} > 0.5\,\text{km} & (f) \\ \sum_n x_{mn} = 1 \quad \forall m \text{ on } S_{OD} & (g) \\ NTR > 50\% & (h) \\ O, D \in F & (i) \end{cases} \tag{1}$$

where,

$D_{OD} =$ direct traveler density of the route starting from O and ending at D;
$F =$ collection of the origin and destination stops;
$O =$ origin stops; $D =$ destination stops;
$S_{OD} =$ the alternative bus route between O and D;
$i, j, m, n =$ stop number;
$SP_{ij} =$ the number of direct travelers between stops i and j;
$L = \sum_{i \in N} \sum_{j \in N} \Delta_{ij} l_{ij} x_{ij} =$ length of the calculated route;
$L_{min}, L_{max} =$ minimum and maximum length of the laid route;
$l_{ij} =$ length of the road section between i and j;

$$x_{ij} = \begin{cases} 1 & \text{if stops } i, j \text{ are on the route } S_{OD} \\ 0 & \text{otherwise.} \end{cases}$$

$$\Delta_{ij} = \begin{cases} 1 & \text{if section } (i, j) \in S_{OD} \\ 0 & \text{otherwise.} \end{cases}$$

$q^x, q^x_{max} =$ nonlinear rate and the maximum nonlinear rate;
$Q^{ij} = SP_{ij} =$ the number of direct travelers on section (i, j);
$Q^{ij}_{max} =$ maximum number of travelers of section (i, j);
$Q^{sum} = \sum_{i \in N} \sum_{j \in N} SP_{ij} x_{ij} =$ total number of travelers of the calculated route;
$Q_{min} =$ the minimum number of travelers of the laid route;
$b^n, b^n_{max} =$ section nonbalance factor and maximum nonbalance factor;
$NTR =$ nontransfer ratio of the entire network.

(*a*) *Length constraint:* rational length of a route is 30–40 minutes per single trip (Wang and Yang, 2001).
(*b*) *Minimum number of direct travelers constraint*: To ensure the efficiency and benefits of the transit companies, the minimum number of direct travelers on one laid route has been set.
(*c*) *Nonlinear constraint:* The factor should be less than 1.5 and is defined as:

$$q^x_{OD} = L_{OD} / P^s_{OD} \tag{2}$$

where, P^s_{OD} is the shortest road distance between the origin O and the destination D.
(*d*) *Section nonequilibrium constraint:* The section nonequilibrium factor is the largest proportion of the maximum section flow to that of the average section flow. It should be less than 1.5 and is defined as:

$$b^n = \frac{\max(Q^{ij} x_{ij})}{Q^{\text{sum}} / (COUNT - 1)} \tag{3}$$

where, $COUNT$ is the number of stops on the laid route.
(*e*) *Section traveler constraint:* The number of section travelers of the laid route must be less than the section capacity that is determined by bus capacity and minimum headway.
(*f*) *Station spacing constraint:* According to the suggested value in the *Code for Transport Planning on Urban Road*, the average stop space $l_{ij} = 0.5$–0.6 km.
(*g*) *Trend constraint:* Because the loop route is not considered, the same route should not pass the same stop twice or more than twice.
(*h*) *Network passenger nontransfer ratio constraint:* Network passenger nontransfer ratio (NTR) is used for the minimum direct ratio. Once the transit network is established, the actual nontransfer ratio can be identified. The equation is:

$$\text{NTR} = \frac{\text{Total number of direct travelers}}{\text{Total traveler O} - \text{D of the planned area}} \tag{4}$$

If there is a relatively big discrepancy between the calculated NTR and the adopted one, the network should be redesigned with the calculated one, until the adopted NTR approximates the calculated one.

3 ALGORITHM FOR THE MODEL

3.1 Parallel Ant Colony Algorithm

ACA is a typical example of applying swarm intelligence to solve combinatorial optimization problems. Dorigo et al. (1996) applied the ACA to solve the TSP and nonequilibrium TSP, QAP, and JSP. Stützle and Hoos (1999) proposed the Max-Min ant system (MMAS), which improved the fundamental ant algorithm in three aspects: (1) The initial pheromone value is set to its maximum value, $\tau_{\max}$, to foster a more adequate optimization search; (2) Only those ants who modify their shortest path after a cycle can alter or add the pheromone; and (3) To avoid prematurely converging the overall optimal solution, the pheromone density of each path is constrained within $[\tau_{\min}, \tau_{\max}]$.

Gambardella and Dorigo (2000) put forward a hybrid ant system, in which the ants found their own solutions in every cycle and used these solutions as the starting points to search for local optimal solutions by using some local search algorithm. Botee and Bonabeau (1998) further studied the selection of parameters with genetic algorithms. Wu et al. (1999) put forward an ACA with mutation features by introducing mutation into the fundamental ACA and utilizing the conciseness and effectiveness of the 2-exchange method.

The principles of the algorithm can be illustrated by examining the food searching process of an ant colony. Along their way from the food source to the nest, ants communicate with one another by means of pheromone. As the ants move, a certain amount of pheromone is deposited on their path and the pheromone gradually evaporates. The ants, then, determine their movements by judging the pheromone density on a path. This process can be described as a loop of positive information feedback, in which the more the ants follow a given trail, the more the pheromone is left, and the higher the probability that this trail will be followed by other ants. This selection process is the result of the ant's self-catalyzing activities, in which the ants can find the optimal path to the final destinations in the end.

Because ACA is an inherently parallelizable search technique on a number of levels, the parallelization of ACA can improve its efficiency and running speed. Because a parallel ACA may be run with a group of computers, the coarse-grained strategy is adopted here. Coarse-grained parallelization schemes run several subcolonies in parallel. The information exchange among the subcolonies is done at certain intervals (*epoch*) or numbers of iterations (Bullnheimer et al., 1998; Middendorf et al., 2002). The key problems in implementing different subpopulations are when, which, and how information should be exchanged among the subcolonies (Middendorf et al., 2002). Recently, some applications were developed in the broader field of parallel computing in transportation (Florian and Gendreau, 2001).

3.2 CPACA for *UBND*

The main idea of bus network design is to find the optimal pair of the origin and destination stops. Different parings can form different bus routes with different direct traveler densities (D_{OD}). It is very similar to the ACA. If we take the buses as *ant colony*, the origin as the *nest*, and the destination as *food source*, the *UBND* problem can be simplified as a process by which the *ant colony* searches for *food* from the *nest* based on the pheromone, that is, searching for an optimal bus route from the origin to the destination based on direct traveler density. The specific steps of the algorithm are as follows.

Step 1 Initialization

- *Bus network initialization:* The bus network consists of bus stops and links, while bus stops can be divided into terminals and middle ones. The links between stops can be either unidirectional or bidirectional. The initial bus network can be expressed as $G = (N, L)$ (N: stops, L: links). Terminals and stops are identified from the collection, and the travel demand between stops is assigned to links in the graph. For those stops evidently unfeasible, the number of travelers can be adjusted to 0 beforehand.
- *CPACA initialization:* The ant colony is initialized by generating m sub-ant colonies and identifying their topological structures, transition time intervals (*epoch*), and scales (n_m), followed by initializing the communication environment (Message Passing Interface, MPI) between the sub-ant colonies.
- *Pheromone matrix initialization:* As all of the initial stops make the same appeal to the bus, an initial weight needs to be allocated to all links. Because the direct traveler density is regarded as the "pheromone," we use the average direct traveler density ($\bar{\tau} = \sum_{i,j\in N} SP_{ij} / \sum_{i,j\in N} l_{ij}$) to initialize the pheromone matrix. Finally, the pioneer buses are assigned to the graph. Because each path carries the same amount of pheromone, the bus can be randomly allocated to the nodes nearby the nest (origin stop).

Step 2 Choosing feasible terminal pairs

Before the path search, the feasible origin O and destination D should be chosen. An OD stop pair is regarded as nonviable if it lies on a route that has already been laid out or does not satisfy the constraints. Then, each sub-ant colony starts searching routes between the OD pair independently.

Step 3 Route searching

- *Choosing feasible stops:* Stops within the constrained distance (0.5–0.8 km) of the current stops k are firstly chosen to form a collection, and the next stop l is then chosen from the collection based on the transition rule.
- *Choosing the next stop:* In the phase, next stops are decided by sequentially choosing feasible entries from the feasible stops. The decision-making about the next stop is based on a probabilistic rule (transition rule) taking into account both the pheromone density τ_{ij} and the visibility value η_{ij} of the corresponding links. Here, τ_{ij} is derived by updating the pheromone on each link after each cycle, based on the transition rule as discussed below; while η_{ij} is derived from a greedy method, which encourages the bus to visit the locally optimal path, here $\eta_{ij} = SP_{ij}/l_{ij}$.
- The value of η_{ij} is fixed in every cycle, moreover, in the *UBND* problem, the links between the stops are asymmetric, that is, SP_{ij} does not always equal SP_{ji}. Therefore, we adopt a directed "pheromone," that is, τ_{ij} does not equal τ_{ji}. To make the searching efficient, we assume that the ants are direction sensible and are able to sense the position of the food. A new local searching rule is then given to the ant colony, which believes that the ants tend to choose the shorter path, that is, they only choose a stop with a shorter distance to the food when compared to the current stop. We define the probability for a bus to move from stop i to stop j as:

$$p_{ij}(k) = \begin{cases} \dfrac{\tau_{ij}^{\alpha} \times \eta_{ij}^{\beta}}{\sum\limits_{h \notin tabu_k} \tau_{ih}^{\alpha} \times \eta_{ih}^{\beta}} & \text{if } j \notin tabu_k \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

  where, $p_{ij}(k)$ = the probability of choosing the j_{th} stop for the k_{th} bus; τ_{ij} and η_{ij} = the pheromone concentration and the visibility on the link (i, j), respectively; α and β = the relative influence of the pheromone trails and the visibility values, respectively; and $tabu_k$ = the set of the infeasible nodes for the k_{th} bus.
- *Calculating the inter-stop travelers:* As Figure 1 shows, for inter-stop direct travelers we count both the demand between the two stops and the demand between the two service areas of both stops. Service area X_k contains stops within walking distance to stop k, as $X_k = \{k, k_1, k_2\}$, and service area Y_l contains stops within walking distance to stops l, as $Y_l = \{l, l_1\}$. Thus, the number of direct travelers from stop k to l is equal to the number of travelers from X_k to Y_l, that is, $SP_{X_kY_l} = SP_{kl} + SP_{kl_1} + SP_{k_1l} + SP_{k_1l_1} + SP_{k_2l} + SP_{k_2l_1}$.
- *Calculating numbers of section travelers:* Section travelers mean the sum of the number of travelers passing through a road section. To calculate the number of travelers of section (k, l), Q^{kl}, we need to sum the number of travelers from X_k to Y_l, and the number of travelers from X_k to Z_m, as illustrated in Figure 2 ($Q^{kl} = SP_{X_kY_l} + SP_{X_kZ_m}$).
- *Calculating the total number of travelers of a route:* $Q^{\text{sum}} = \sum\limits_{(k,l)\in S_{OD}} Q^{kl}$;
- *Calculating route length:* $L = \sum\limits_{i\in N} \sum\limits_{j\in N} \Delta_{ij} l_{ij} x_{ij}$;
- *Calculating route direct traveler density:* $D_{OD} = Q^{\text{sum}}/L$.

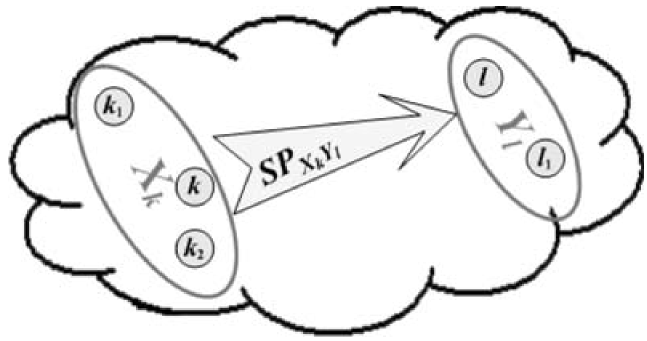


Fig. 1. The number of direct travelers between inter-stops.

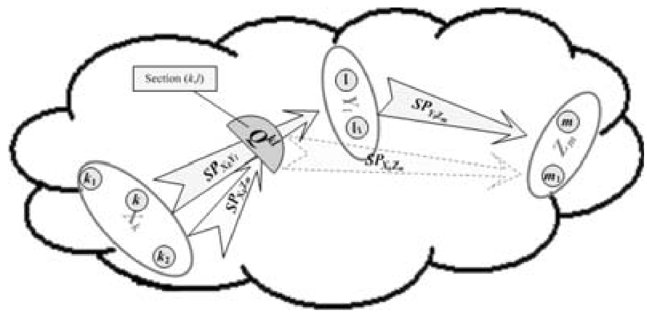


Fig. 2. The number of direct travelers of road sections.

Each route is evaluated in turn after the calculation. If the routes satisfy the *length constraint*, the *minimum number of direct travelers constraint*, and the *nonlinear constraint*, the bus network will be updated accordingly; otherwise, the route will be abandoned and the algorithm returns to Step 2.

Step 4 The pheromone update rule

After a route has been identified, the pheromone in the network needs to be updated.

- *Pheromone assignment rule:* While finding a route, the network optimization is fully dependent on the guidance of pheromone τ_{ij} and visibility η_{ij}. The visibility η_{ij} is comparatively stable, so the pheromone τ_{ij} becomes particularly important in terms of searching for new routes. The most common strategy for calculating the increased pheromone is the Ant-Circle method (Badr and Fahmy, 2004), which incorporates global information. However, because it neglects local information, its convergence speed and optimal quality are not good. Therefore, with reference to Ant-Density Strategy, a new update strategy, namely Ant-Weight Strategy, is put forth through integrated consideration of the global and local information. Specifically, the update strategy to the increased pheromone is:

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{f^k} \times \dfrac{f^k - f_{ij}}{(n-2)\,f^k} & \text{if link } (i,\, j) \text{ on the } k_{th} \text{ route} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\Delta\tau_{ij}^{k}$ = the increased pheromone on link (i, j) of route k found by the ant; Q = a constant; f^k = the direct traveler density of route k; f_{ij} = the direct traveler density on link (i, j); and n = the number of the stops along route $k (n > 2)$.

The strategy for updating the increased pheromone consists of two components: the first is the global pheromone increment Q/f^k related to the direct traveler density of route k; the second is the local pheromone increment $\frac{f^k - f_{ij}}{(n-2)\,f^k}$ based on the contribution of link (i, j) to route k. Thus, the update strategy ensures that the assigned pheromone increments are directly proportional to the direct traveler density. The more favorable the link/route, the more the pheromone increment allocated to it, and the more the accurate directive information is provided for later search. Meanwhile, by adjusting the pheromone assigning method for the links of current optimal path automatically, the algorithm can facilitate more delicate searches in the next cycle in a more favorable area. This helps to improve the learning capacity of the algorithm from past searches, and enhances the efficiency. In addition, to avoid local optimization and to enlarge the probability of obtaining a higher-quality solution, upper and lower limits $[\tau_{\min}, \tau_{\max}]$ are fixed to the updating equation. Therefore, the pheromone of a link after a cycle can be updated by:

$$\tau_{ij}^{\text{new}} = \rho \times \tau_{ij}^{\text{old}} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \quad \rho \in (0, 1) \tag{7}$$

$$\tau_{\min} = \min(f_{ij}),\, \tau_{\max} = \max(f_{ij}) \tag{8}$$

$$\tau_{ij} = \begin{cases} \max\left(\tau_{\min},\, \tau_{ij}^{\text{new}}\right) & \tau_{ij}^{\text{new}} < \bar{\tau} \\ \min\left(\tau_{\max},\, \tau_{ij}^{\text{new}}\right) & \tau_{ij}^{\text{new}} \geq \bar{\tau} \end{cases} \tag{9}$$

where τ_{ij}^{new} = the pheromone on the link(i, j) after updating;τ_{ij}^{old} = the pheromone on the link(i, j) before updating; and ρ = the constant that controls the speed of evaporation.

- *The immigration rule:* In “ant immigration,” some outstanding ants are moved to other colonies after some number of searches to stimulate the ants to find more favorable paths. The topology of connection between the sub-ant colonies needs to be identified before the immigration. Unidirectional ring topology is adopted, which means that subcolony r exchanges individually with subcolony $r + 1$ at certain intervals or numbers of iterations to connect the subcolonies in this study. Then, the search of an ant is completed, the algorithm returns to Step 3 to let other ants search. The iteration will not stop until all ants of the subcolonies have completed the searches.

Step 5 Generating alternative routes

After all ants have completed a cycle, the route with the largest direct traveler density between the *OD* is chosen to be an alternative route. The stops it covers are sequentially added into S_{OD}. Thus, the search of routes between the *OD* is completed, and the algorithm returns to Step 2 for the next *OD* route search. This process repeats till all valid routes between *ODs* are found. In this way, no more than one alternative route is generated between a valid *OD*. The collection of the routes is written as ***S***.

Step 6 Laying routes

The route with the largest density in collection ***S*** is then chosen to be added into the network, with the others abandoned, after the relevant data are updated.

Step 7 Revising the network

After a route layout, the number of travelers carried by the route needs to be subtracted from the original travelers matrix. First, the total number of travelers of each route section, and its capacity, are calculated. If the capacity of every section of a route outweighs the flow, all passengers on the route should be carried, and the number of inter-stop travelers should be subtracted from the traveler matrix. If the section capacity is less than the flow, only the number of travelers being carried is subtracted from the matrix.

- *Route capacity calculation:* The route capacity is the maximum number of travelers that can be carried by a route. It is determined by the largest number of bus vehicles that can be served by a stop (Wang and Yang, 2001). The route capacity is: ($C_1 = R \times r \times N_1 \times i \times K_i$), here $R =$ bus passenger capacity; $r =$ load factor (0.85 for peak hours and 0.5 for off-peak hours); $N_1 =$ stop capacity with only one homonymous stop; $i =$ the number of homonymous stops ($i \leq 3$); and $K_i =$ the utilization factor of the homonymous stops ($K_i = 1.0$, when $i = 1$; $K_i = 0.8$, when $i = 2$; $K_i = 0.7$, when $i = 3$). Here, the bus route is laid seriatim, and a loop route is not considered; only one homonymous stop can be placed onto one route, that is, $i = 1$, $K_i = 1$.

  It is notable that not all passengers involved in calculating the route capacity are direct travelers but might be transfer ones. Hence, to deduce the route capacity of direct travelers, the number of transfer ones should be deducted, namely, to multiply C_1 by the nontransfer ratio (B_w). Assume that the ratio is 50%, the capacity (direct travelers only) of a single route is as: $C_1 = R \times r \times N_1 \times i \times K_i \times B_w$. The nontransfer ratio should be known beforehand, however, it is unknown before the network is established. Therefore, an initial nontransfer ratio may be determined by the current conditions and recalculated after the network has been identified. If there is a large discrepancy between them, the latter should be used to redesign the network, until the two come within a close range.
- *Residual travelers identification:* Because section flow is generated from the total number of travelers passing through a section, there should be some residual travelers when the section is overloaded. At a stop, any passengers may get on a bus and any of them may also be uncarried. Therefore, an *OD* pair that passes an overloaded section (k, l) may generate a residual volume on that section as:

$$T^1{}_{ij}^{kl} = \Delta C_{kl} \cdot T_{ij} y_{ij}^{kl} / \sum_{i \in N, j \in N} T_{ij} y_{ij}^{kl} \tag{10}$$

$$\Delta C_{kl} = V^{kl} + Q^{kl} - C_2 \tag{11}$$

$$y_{ij}^{kl} = \begin{cases} 1 & \text{if pair } (i,\ j) \text{ pass through section } (k, l) \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

  where $T^1{}_{ij}^{kl}=$ the residual travelers between pair (i, j) passing through the overloaded section (k, l); $T_{ij} =$ travelers between pair (i, j); $\Delta C_{kl} =$ the overload travelers in the overloaded section (k, l); $V^{kl} =$ the sum of travelers of the overloaded section (k, l) on the previously laid routes ; $Q^{kl} =$ current added travelers on the overloaded section (k, l); and $C_2 =$ the route capacity.

  All stop pairs passing the overloaded section can generate some corresponding left flow. To ensure that none of the sections along the route are overloaded, we take the maximum left flow of each section for every stop pair on the route, that is, $T_{ij}^1 = \max[T^1{}_{ij}^{kl}]$, where $T_{ij}^l =$ the left flow of a stop pair (i, j) on the route. The travelers on the route are $T_{ij}^2 = T_{ij} - T_{ij}^1$. The left flow is zero for the stops that do not pass the overloaded sections.
- *Travelers matrix revision:* After identifying the stop left flow, the carried and left flows can be calculated. The carried flow of a route is deducted from the original flow matrix, and is then assigned to the route. The newly added section flow Q^{kl} can be recalculated and added upon the section flow V^{kl}. The total section flow is hitherto $V_{kl} + Q^{kl}$, and is less than the corresponding capacity. It is reserved as the existent section flow V^{kl} for laying the next route.

  So far, the route with the largest flow density in this cycle is laid to the network. The cycle completes after the revision of the network. The next step is to determine whether to terminate or not.

Step 8 Terminating condition

If there is no route satisfying the constraints or the iteration reaches its maximum, stop; otherwise return to Step 2. Figure 3 illustrates the entire flow of the algorithm.

4 NUMERICAL TEST

This model was tested with two sets of data. The first one attempted to illustrate the validity of the model on a simple network; and the second set aimed to test the performance of the model and an algorithm for designing a real bus network in Dalian, China.

4.1 Test 1

A sample network with 17 nodes and 29 links is defined in Figure 4. A symmetrical hourly demand matrix is randomly generated, as shown in Table 1. To evaluate the performance of the *maximum direct traveler density* method (MDTD), both the *maximum direct*

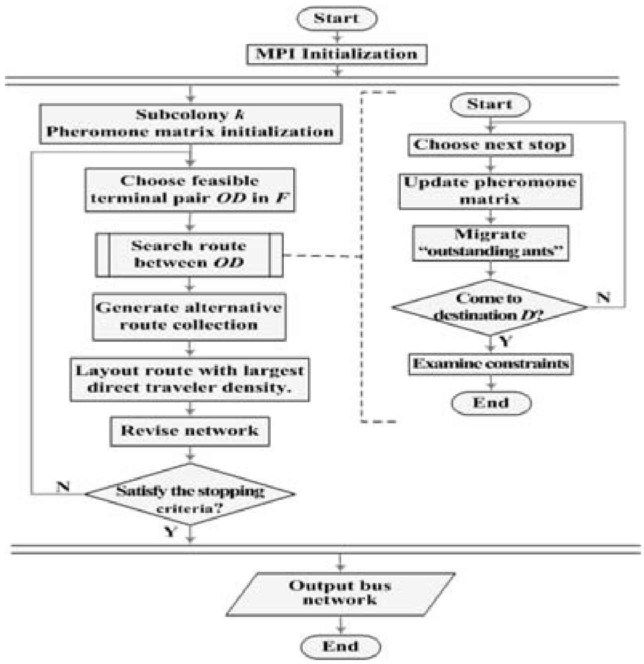


Fig. 3. The structure of the algorithm proposed in this study.

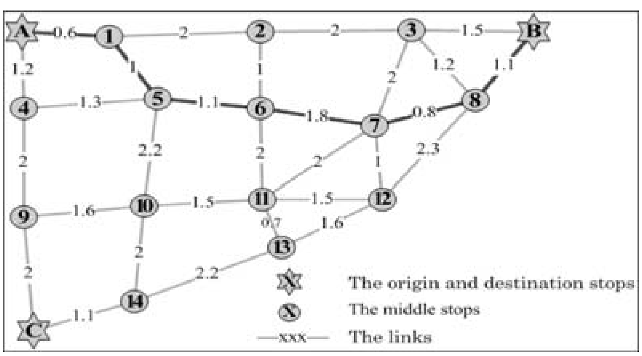


Fig. 4. Description of example network.

traveler method (MDT) and the *maximum direct travelers on shortest paths* method (MDTSP) are used, respectively, to design the route with the data. According to their objectives, different routes between each *OD* are found, and the corresponding numbers of *direct travelers*, *lengths*, and *direct traveler densities* are calculated to get Table 2.

It can be seen that the first laid route is different due to the method used, for example, with MDTD the first laid route is *A*-*B*, and with MDT the first laid route is *B*-*C*, while with MDTSP the first laid route is also *B*-*C*. From Table 2, it is can be seen that the *direct traveler density* of the route with MDTD is the largest. This is just as expected because MDTD has jointly considered the number of the direct travelers and the length of the route, thus gaining the best result. It also can be said that the solution of MDT is better than that of MDTSP because it has been released from the limitation of the shortest paths and because the searching space has been widened.

Table 1
Demand matrix

	A	*B*	*C*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
A	0	3	4	4	7	9	5	8	8	9	10	9	6	2	6	4	5
B	3	0	0	0	5	7	6	3	3	5	9	10	7	7	4	0	8
C	4	0	0	3	9	1	7	8	3	8	1	4	7	5	5	4	7
1	4	0	3	0	2	5	8	7	9	1	8	10	0	6	6	5	4
2	7	5	9	2	0	4	6	8	3	5	9	6	1	1	4	5	2
3	9	7	1	5	4	0	0	9	9	7	9	5	8	3	7	5	9
4	5	6	7	8	6	0	0	4	1	5	7	5	4	10	7	6	6
5	8	3	8	7	8	9	4	0	8	6	5	6	8	7	5	1	8
6	8	3	3	9	3	9	1	8	0	2	9	3	5	3	4	4	4
7	9	5	8	1	5	7	5	6	2	0	4	8	8	1	6	7	9
8	10	9	1	8	9	9	7	5	9	4	0	2	7	3	10	3	5
9	9	10	4	10	6	5	5	6	3	8	2	0	9	3	5	4	2
10	6	7	7	0	1	8	4	8	5	8	7	9	0	7	1	5	6
11	2	7	5	6	1	3	10	7	3	1	3	3	7	0	4	2	3
12	6	4	5	6	4	7	7	5	4	6	10	5	1	4	0	2	1
13	4	0	4	5	5	5	6	1	4	7	3	4	5	2	2	0	5
14	5	8	7	4	2	9	6	8	4	9	5	2	6	3	1	5	0

4.2 Test 2

To examine the model and the efficiency of the algorithm, data in Dalian city are used for the numerical test. The population in Dalian is about 2 million, and the build-up area is about 180 km^2. Its road network consists of 3,200 links and 2,300 nodes. At present there are 89 bus routes that extend 1,130 km and include 1,500 bus stops. We get the bus *OD* traffic matrix through on-board surveys of the entire 89 routes. The parameters in the CPACA (Table 3) are estimated through simulation. There are eight ant colonies, and each has 30 ants. The model and algorithm are carried out with Microsoft Visual C^{++}. Net 2003 on the cluster environment is formed by eight computers.

In total, 82 bus routes are laid out with our method, extending 950 km. By comparing the optimized bus network with the present one, it is known that the direct traveler ratio of the present bus network is about 28.8 persons/km, which is much lower than the optimized 34.3 persons/km (Figure 5). This is mainly because present bus routes overlap each other to an extent that can disperse the trip flow, thus lowering the efficiency of the network. In an optimized case, direct trips share 61% of all trips, while the present one is about 41%. As Figure 6 shows, the lengths of the optimized bus routes concentrate mainly in 5–15 km; however, the long routes (>15

Table 2
Routes for three approaches

	Origin and destination	*Route description*	*Direct travelers*	*Length*	*Direct traveler density*
Maximum direct traveler density	***A-B***	***A-1-5-6-7-8-B***	***121***	***6.4***	***18.91***
	A-C	A-1-5-10-14-C	85	6.9	12.32
	B-C	B-8-7-12-11-10-14-C	148	9	16.44
Maximum direct travelers	A-B	A-1-5-6-7-3-B	122	8	15.25
	A-C	A-1-5-10-14-C	85	6.9	12.32
	B-C	**B-3-7-6-5-10-14-C**	**173**	**11.7**	**14.79**
Maximum direct travelers on shortest paths	A-B	A-1-2-3-B	46	6.1	7.54
	A-C	A-4-9-C	34	5.2	6.54
	B-C	**B-8-7-12-13-14-C**	**103**	**7.8**	**13.21**

Table 3
Parameters in CPACA

m *(the number of subcolonies)*	p *(the population of subcolony)*	α *(a constant)*	β *(a constant)*	Q *(a constant)*	*epoch (migrating intervals)*	n_m *(the number of migrating ants)*
8	30	2	1	1,000	10	1

km) in the present bus network exceed 30% of the total. And the nonlinear rates of the optimized routes are around 1.5, lower than the present ones.

At last, we try to compare CPACA with MMAS (Stützle and Hoos, 1999) and with ACA + (ACA with the Ant-Weight strategy). We test 10 times for the three algorithms with the same data. From Figure 7, we can see that the *average direct traveler density* of CPACA is the largest. This may be because the introduction of the migration operation in CPACA diversified the ant colony and widened the searching space. As the result, CPACA can prevent the algorithm from trapping in local optimization, and it improves the optimizing quality. Also, the ACA + generally gives a better solution than the MMAS. This indicates that the Ant-Weight strategy is effective compared to MMAS. Moreover, the run time depends not only on the CPU but also on the operation system, the compiler, the programming language, and the decided accuracy in the calculation. Thus, a fair comparison of computational efficiency is difficult. However, it is pretty obvious that the computational time of CPACA is much less than that of others. Thus, we can conclude that CPACA can improve the optimizing quality and save computing time greatly.

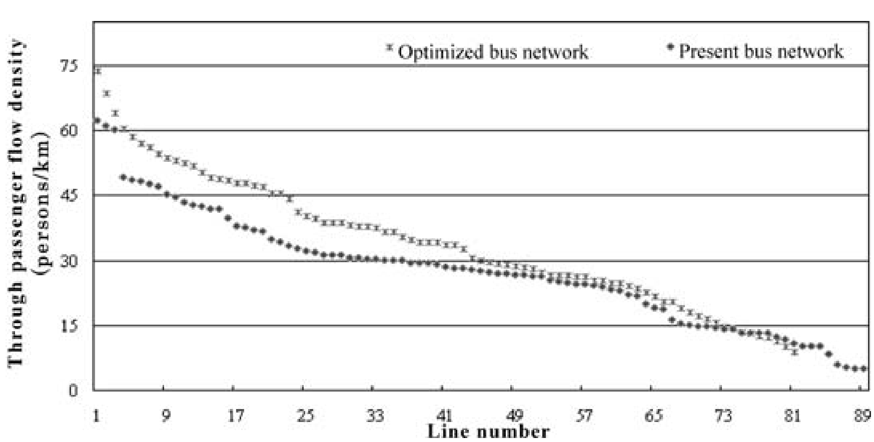
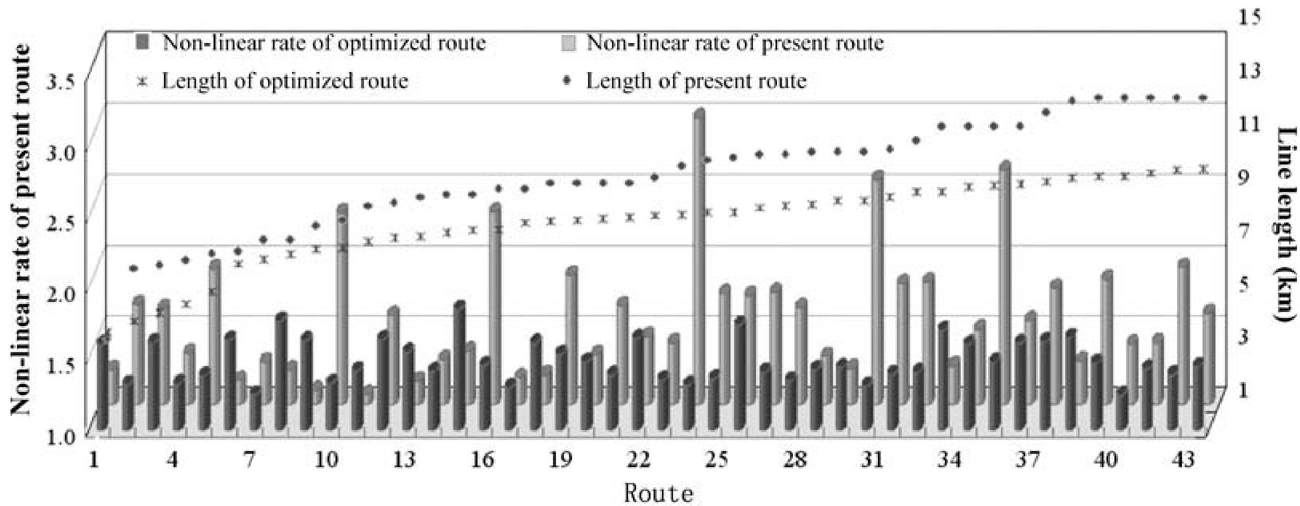
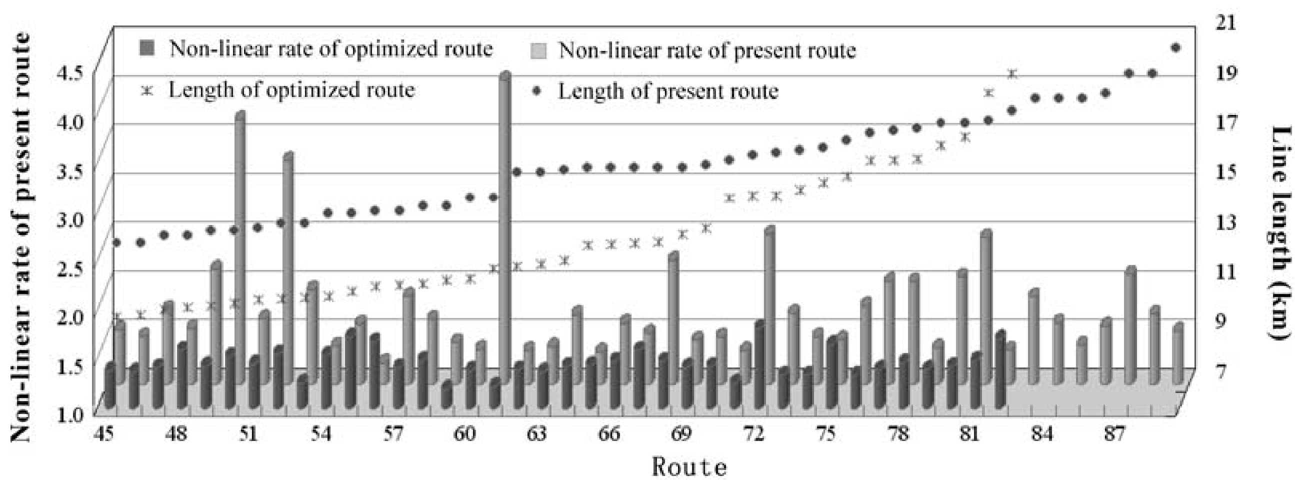


Fig. 5. Comparisons of direct traveler densities.



(a)



(b)

Fig. 6. Comparisons of lengths and non-linear rates.

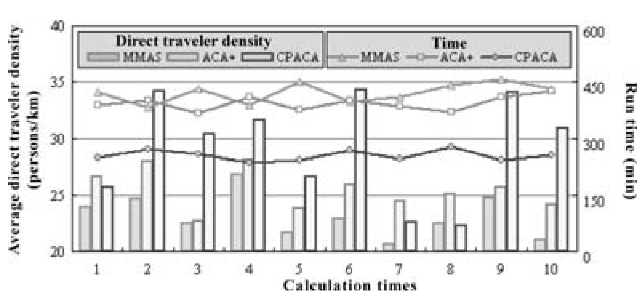


Fig. 7. Comparisons of average direct traveler density and computational times.

5 CONCLUSIONS

Traditional direct traveler approaches are practical and convenient. Because they usually set the route along the shortest path and do not consider the length of the route; however, there are some limitations in the feasibility of practical application and design results. A new model for optimizing urban bus networks has been developed here, which takes maximum direct traveler density as an objective. The model aims to minimize the average trip time, namely to allow as many passengers as possible to travel between starting points and destinations without transfers. On the other hand, the model maximizes profits for bus companies, namely by increasing operation efficiency and shortening the total lengths of bus routes. This study identifies some of the features of the *UBND* that make it difficult for traditional methods to solve. We propose an ant colony algorithm, together with an evolutionary optimization mechanism, as a tool to solve the problem. To improve the optimization and to quicken the calculation, CPACA is adopted. This study shows that a coarse-grain model with less communication is more suitable in a cluster environment. The algorithm comprises several sub-ant colonies and completes the communication between the sub-ant colonies by MPI. All of the sub-ant colonies maintain their own pheromone matrix, and they independently search for the optimal bus route between the origin and the destination according to the "pheromone," that is, the direct traveler density, along the given network. Furthermore, the route with the largest direct traveler density among all alternative viable routes is chosen as the route to be laid in a cycle. Finally, with data from Dalian City we test our model and compare some indices between the optimized and present bus networks. Results show that the CPACA-based procedures are successful in solving the bus route design problem.

Improvement of the searching efficiency will be our future goal. We will further our research from two aspects: (1) Simplifying the initial network; we will adopt a platform, which covers one or several close stops, to simplify the whole path network and to improve the algorithm's efficiency. By simulation, we have found that while the number of stops was less than 1,000, the stability and runtime of the algorithm was much more satisfactory. (2) Improving efficiency of the CPACA; we will first improve the searching efficiency of ACA and decrease the account of invalid searching, which involves selecting the right pheromone updated strategy and further optimizing the parameters in ACA. The simulation experiments showed that the opposed Ant-Weight strategy could accelerate convergence to some extent, but a better solution can be obtained by using dynamic pheromone updated strategy. We also need to reduce the communication between subcolonies. Moreover, this study aims to maximize the direct traveler density and to generate only one route in each cycle, which had affected the route design. Thus, the final solution may not be the global optimal, and some areas with less passenger density may become unserved areas.

ACKNOWLEDGMENTS

This research is financed by the National Natural Science Foundation of China through project 50422282 and the Doctoral Base Foundation of Chinese Education Ministry through project 20050151007.

REFERENCES

Baaj, M. H. & Mahmassani, H. S. (1995), Hybrid route generation heuristic algorithm for the design of transit networks, *Transportation Research*, *Part C*, **3**(1), 31–50.

Badr, A. & Fahmy, A.(2004), A proof of convergence for ant algorithms. *Information Sciences*, **160**, 267–79.

Botee, H. M. & Bonabeau, E. (1998), Evolving ant colony optimization, *Complex Systems*, **1**(2), 149–59.

Bullnheimer, B., Kotsis, G. & Strauss, C. (1998), Parallelization strategies for the ant system, in R. Leone, A. Murli, P. Pardalos, and G. Toraldo (eds.), *High Performance Algorithms and Software in Nonlinear Optimization*, Kluwer, Dordrecht, pp. 87–100.

Chakroborty, P. (2003), Genetic algorithms for optimal urban transit network design, *Computer-Aided Civil and Infrastructure Engineering*, **18**, 184–200.

Chakroborty, P., Deb, K. & Subrahmanyam, P. S. (1995), Optimal scheduling of urban transit system using genetic algorithms, *ASCE Journal of Transportation Engineering*, **121**(6), 544–53.

Dashora, M. & Dhingra, S. L. (1998), Expert system for bus routing and scheduling, in P. K. Sikdar, S. L. Dhingra, and K. V. Krishna Rao (eds.),Vol. 2. *Computers in Urban Planning and Urban Management*, Narosa Publishing, New Delhi, India, pp. 841–53.

Dorigo, M. (1992), Optimization, learning and natural algorithms, Ph.D. thesis, Politecnico di Milano, Italy.

Dorigo, M., Maniezzo, V. & Colorni, A. (1996), The ant system: Optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man and Cybernetics*, **1**(26), 29–41.

Dubois, D., Bel, G. & Libre, M. (1979), A set of methods in transportation network synthesis and analysis, *Journal of Operational Research Society*, **30**(9), 797–808.

Fernandez, R. (1993), An expert system for the preliminary design and location of high capacity bus stops, *Traffic Engineering Control*, **34**, 533–39.

Florian, M. & Gendreau, M. (2001), Applications of parallel computing in transportation, *Parallel Computing*, **27**, 1521–22.

Gambardella, L. M. & Dorigo, M. (2000), An ant colony system hybridized with a new local search for the sequential ordering problem, *INFORMS Journal on Computing*, **12**(3), 237–55.

Garey, M. R. & Johnson, D. S. (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco.

Guan, J. F., Yang, H. & Wirasinghe, S. C. (2004), Determine network transit line configuration for optimal passenger line assignment, in *Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control*, **2**, 839–43.

Hasselström, D. (1981), Public transportation planning—A mathematical programming approach, Doctoral dissertation, University of Göteborg, Sweden.

Michael, R. B., Peter, K. & Uwe, T. Z. (1997), Optimal lines for railway systems, *European Journal of Operation Research*, **96**(1), 54–63.

Middendorf, M., Reischle, F. & Schmeck, H. (2002), Multi colony ant algorithms, *Journal of Heuristics*, **8**, 305–20.

Senevirante, P. N. (1990), Analysis of on time performance of bus services using simulation, *Journal of Transportation Engineering*, **116**(4), 517–31.

Simonis, C. (1981), Optimierung von omnibuslinien. Berichte stadt–region–land, Institut für Stadtbauwesen, RWTH Aachen.

Sonntag, H. (1979), Heuristic procedures for planning demand-oriented traffic routes in public transport systems, *Z. Operation Research Series A-B*, **23**, B15–31.

Steenbrink, P. A. (1974), *Optimization of Transport Networks*, Wiley, London.

Stützle, T. & Hoos, H. (1999), Max-min ant system and local search for combinatorial optimization problems, in S. Voß, S. Martello, I. H. Osman, and C. Roucairol (eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic, Boston, MA, pp. 313–29.

Su, C. T. & Weng, R. C. (2003), Sloving TSP Problems by ant colony optimization and noising method, *Commerce & Management Quarterly*, **4**(4), 359–75.

Wang, W. & Yang, X. (2001), *Planning and Management Technology for Urban Transit Systems*, Science Press, Beijing.

Wu, Q. H., Zhang, J. H. & Xu, X. H. (1999), An ant colony algorithm with mutation features, *Journal of Computer Research & Development*, **36**(10), 1240–45.

Yin, P. Y. (2003), Ant colony search algorithms for optimal polygonal approximation of plane curves, *Pattern Recognition* **36**, 1783–97.