DESIGNING BUS ROUTES AND FREQUENCIES FOR TIN SHUI WAI, HONG KONG

W.Y. Szeto,* Yongzhong Wu, S.C. Wong
Department of Civil Engineering, The University of Hong Kong
E-mail: ceszeto@hku.hk

Abstract: A real bus network design problem for a suburban residential area, Tin Shui Wai, Hong Kong, is investigated. The problem considers bus service from origins inside this area to destinations in the city. The aim is to improve the existing bus network by reducing the number of transfers and total travel time of users. This is achieved by the proposed integrated solution method, which simultaneously solves the route design and frequency setting problems. In the proposed method, a genetic algorithm that tackles the route design problem is hybridized with a neighborhood search heuristic that addresses the frequency setting problem. A new representation scheme and specific genetic operators are developed so that the genetic algorithm can search all possible route structures rather than selecting from an initial set of predefined routes. The proposed method reduces the number of transfers and total travel time by 20.6% and 7.0%, respectively.

Keywords: Bus network design; Route design problem; Frequency setting problem; Genetic algorithm, Neighborhood search

1. INTRODUCTION

With Hong Kong experiencing continuous population growth, new suburban residential areas have emerged on the periphery of the city. The design of the transit networks for these areas is unique to provide adequate transportation services from origins inside the areas to urban working and shopping destinations.

This paper focuses on the bus network design problem for a suburban residential area, Tin Shui Wai (TSW), in Hong Kong. Most of the residents of TSW work in the urban areas of Hong Kong. External bus service is their main commuting mode. The first external bus routes commenced operating in the early 1990s. Since then, external bus routes in TSW have developed with the growth of the town. All of the bus routes exit TSW through the Tai Lam Tunnel (TLT) located on the southeast side of the area and continue on to the highway connected to urban destinations. Free transfer at the TLT bus interchange allows a wide range of journeys to be made among various zones inside TSW and between TSW and destinations in the city.

However, because of a lack of a systematic design, the existing bus network operates in an inefficient manner. Many passengers need to transfer at the TLT interchange. At the same time, some bus services are routed to loop around various zones to provide direct journeys. This results in an increase in travel time. The restructuring of the bus routes in TSW to reduce both the number of transfers and the total travel time under relatively fixed operating costs is the key concern of the bus operator.

The investigated bus network design problem consists of two sub-problems in the sequential transit planning process that are stated in Ceder and Wilson (1986), namely, route design and frequency setting. These sub-problems have been addressed in some work. They are included in the systematic and comprehensive review of transit network design conducted by Guihaire and Hao (2008), while Lampkin and Saalmans (1967) tackle them sequentially, stating that simultaneously choosing routes and frequencies is too great a problem to handle. However, it is arguable that the interaction made possible by dealing with both the route design and frequency setting problems at the same time is beneficial to the final results (Guihaire and Hao, 2008). In addition, the passenger travel time cannot be evaluated without knowing the frequency, which determines the passenger trip assignment and waiting time.

Only a few researchers have tackled the route design and frequency setting problems simultaneously. Fan and Machemehl (2006a, 2006b) consider frequency during the network evaluation process when solving the route network design problem. They propose a genetic algorithm (GA) and simulated annealing algorithm (SA) to minimize the sum of user costs (including waiting, walking, transfer, and in-vehicle travel costs), operating cost (which is proportional to the number of buses), and unsatisfied demand cost. Tom and Mohan (2003) present a genetic algorithm that simultaneously determines routes and frequencies, to minimize both operating and user total travel costs. They propose a coding scheme that incorporates the selection of both routes and frequencies, so that the route design and frequency setting problems can be solved simultaneously. However, like that of Fan and Machemehl (2006a, 2006b), their approach requires an initial set of candidate routes. Although the initial route generation procedure significantly reduces the problem complexity, it reduces the level of the quality of the final solution, and its suitability varies across network settings.

The bus network design problem investigated in this paper simultaneously considers route design and frequency setting, and is characterized by four aspects that distinguish our paper from others in the related literature. First, the problem considers the total travel time and operating cost, both of which are related to frequency. Thus, the objectives of this study are also different from those in the existing literature. Second, the solution method proposed considers all possible route structures with all possible stop combinations. Third, the solution method developed is different from the GAs and hybrid GAs proposed in the literature. Fourth, the layout of the studied route network is distinct. The investigated bus network design involves a suburban area with only one large interchange, which allows a wide range of journeys to be made among various zones inside the suburban area and to destinations in the urban area. To the best of our knowledge, this feature is not found in other transit network design papers.

The remainder of the paper is organized as follows. In the next section, the formulation of the problem is presented. The proposed solution method is described in Section 3. The computational results are given in Section 4, and compared with those of the current network. Finally, the work is concluded in Section 5.

2. PROBLEM FORMULATION

As shown in Figure 1, TSW is divided into 23 zones (shown as nodes 1-23). Free transfer at the TLT bus interchange stop (T) allows a wide range of journeys to be made. There are five destinations for the bus routes to the urban area (shown as nodes 24-28). Each bus route goes to one of the five destinations. The demand matrix *D* and vehicle travel times (in minutes) between nodes (as indicated in Figure 1) are known. This work aims to completely restructure the bus network inside this area to reduce the total number of transfers and total travel time (including waiting time) of users without increasing the number of operating vehicles. The problem can be formulated as follows.

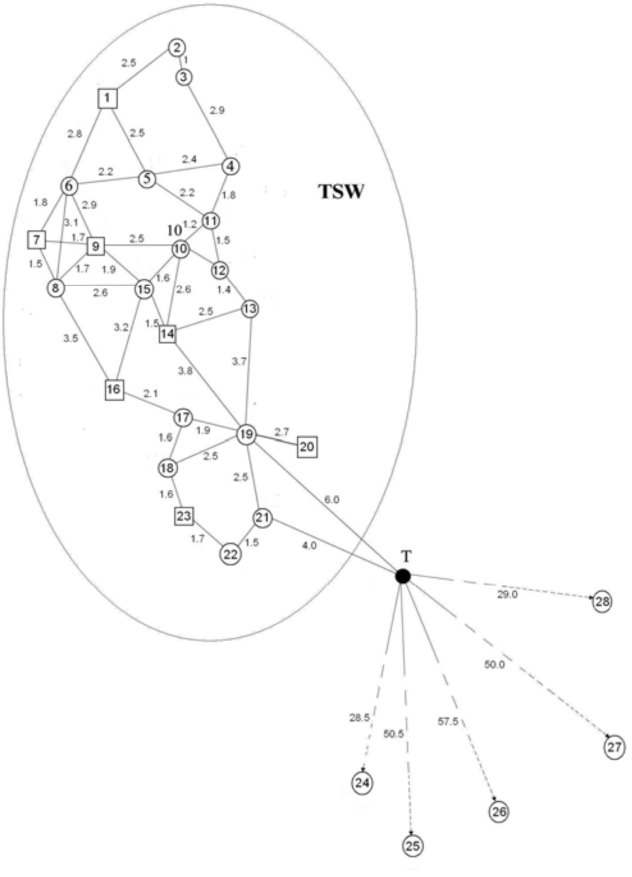


Figure 1. Tin Shui Wai bus network.

$$\min z = B_1 \sum_{i\in Z}\sum_{e\in V} d_{ie} NR_{ie} + B_2 \sum_{i\in Z}\sum_{e\in V} d_{ie} T_{ie} \tag{1}$$

subject to

$$\sum_{j\in U\cup\{0\}} X_{0jn} = 1 \qquad \text{for } n = 1 \text{ to } R_{\max}, \tag{2}$$

$$\sum_{i\in V\cup\{0\}} X_{i0n} = 1 \qquad \text{for } n = 1 \text{ to } R_{\max}, \tag{3}$$

$$\sum_{i\in Z\cup\{0\}} X_{ijn} - \sum_{i\in Z\cup\{0\}} X_{jin} = 0 \qquad \text{for } j \in Z,\ n = 1 \text{ to } R_{\max}, \tag{4}$$

$$\sum_{i\in Z\cup\{0\}} X_{ijn} \le 1 \qquad \text{for } j \in Z,\ n = 1 \text{ to } R_{\max}, \tag{5}$$

$$\sum_{j\in Z\cup\{0\}} X_{ijn} \le 1 \qquad \text{for } i \in Z,\ n = 1 \text{ to } R_{\max}, \tag{6}$$

$$T_n = \sum_{i\in Z}\sum_{j\in Z} X_{ijn}(c_{ij} + s) - s \qquad \text{for } n = 1 \text{ to } R_{\max}, \tag{7}$$

$$\sum_{n=1}^{R_{\max}} 2 f_n T_n \le W , \tag{8}$$

$$f_{\min} \le f_n \qquad \text{for } n = 1 \text{ to } R_{\max}, \tag{9}$$

$$\sum_{i\in Z}\sum_{j\in Z} X_{ijn} \le S_{\max} \qquad \text{for } n = 1 \text{ to } R_{\max}, \tag{10}$$

$$T_{ie} = \begin{cases} \sum_{r_n\in DR_{ie}} f_n T_{ie}^n / \sum_{r_n\in DR_{ie}} f_n + 1/ \sum_{r_n\in DR_{ie}} f_n, & \text{if } |DR_{ie}| \ne 0 \\ \sum_{r_n\in IR_{ie}} f_n T_{ie}^n / \sum_{r_n\in IR_{ie}} f_n + 1/ \sum_{r_n\in IR_{ie}} f_n, & \text{if } |DR_{ie}| = 0 \end{cases} \quad \text{for } i \in Z,\ e \in V, \tag{11}$$

$$T_{ie}^n = \sum_{i\in ND_{ie}^n}^{N} \sum_{j\in ND_{ie}^n}^{N} X_{ijn}(c_{ij} + s) + \sum_{i\in ND_{ie}^n}^{N} \sum_{j\in V}^{N} X_{ijn} c_{ie} \quad \text{for } i \in Z/V,\ e \in V \quad n = 1 \text{ to } R_{\max}, \tag{12}$$

$$RT_{ie}^n = X_{ien} + \sum_{j\in Z/V} X_{ijn} RT_{je}^n \qquad \text{for } i \in Z/V,\ e \in V,\ n = 1 \text{ to } R_{\max}, \tag{13}$$

$$NR_{ie} = \prod_{n=1}^{R_{\max}} (1 - RT_{ie}^n) \qquad \text{for } i \in Z/V,\ e \in V . \tag{14}$$

Notation

a. Sets/Indices

$Z = \{1,2,...,N\}$: set of nodes (including TSW zones and destinations);

U: set of bus terminals inside TSW;
V: set of destinations outside TSW, i.e., nodes 24-28;

i, *j* , *e*: indices of nodes;

r_n: the *n*th route;
DR_{ie} = set of direct routes from node *i* to destination *e*;
IR_{ie} = set of indirect routes from node *i* to destination *e*; and

ND_{ie}^{n} = set of nodes between *i* (including *i*) and *e* (excluding *e*) on the *n*th route.

b. Parameters

N = number of nodes (zones) in the network, i.e., 28;
M = number of destinations outside TSW, i.e., 5;
c_{ij} = in-vehicle travel time on the shortest path between nodes *i* and *j*;
s = average time cost for stopping at a node, i.e., 1.5 minutes;
d_{ie} = travel demand from node *i* to destination *e*;
W = maximum bus fleet size allowed for the network;
R_{max} = maximum number of routes in the bus network, i.e, 10;
f_{min} = minimum frequency of a route;
S_{max} = maximum number of stops (including the bus terminal) within TSW on a route, i.e., 9;
B_1 = weight for the number of transfers; and
B_2 = weight for the total travel time.

c. Decision Variables

X_{ijn} = 1 if the *n*th (n = 1 to $R_{\max}$) route stops at node *j* immediately after node *i*; and 0 otherwise;
X_{0jn} = 1 if the *n*th route starts at node *j*; and 0 otherwise;
X_{i0n} = 1 if the *n*th route ends at node *i*; and 0 otherwise;
RT_{ie}^{n} = l if the *n*th route goes from *i* to destination *e*; and 0 otherwise;
NR_{ie} = l if there is no route going from node *i* to destination *e*; and 0 otherwise;
f_n = frequency of the *n*th route;
T_n = single trip time of the *n*th route;
T_{ie}^{n} = travel time going from zone *i* to destination *e* via the *n*th route; and
T_{ie} = average travel time of passengers from zone *i* to destination *e*.

The objective of the problem is to minimize the weighted sum of the number of transfers and network travel time (including in-vehicle travel time and waiting time) subject to the constraint on the maximum fleet size, maximum number of stops on each route, and minimum frequency for each route.

A dummy node 0 is considered in the network. With this dummy node, X_{i0n} = 1 means that the *n*th route ends at node *i*, while X_{0jn} = 1 means that the *n*th route starts from node *j*. Constraint (2) ensures that all routes start from a terminal selected from the available locations (shown as squares in Figure 1). Constraint (3) ensures that each route has a destination selected from the available destinations. It should be noted that the *n*th route is not selected in the network when X_{00n} = 1. Constraint (4) ensures that except dummy nodes, any node on a route has one preceding and one following node. Constraints (5) and (6) ensure that each node can be visited by a particular route only once. Constraint (7) calculates the in-vehicle travel time (including stop time) of a route. Constraint (8) ensures that the fleet size cannot exceed the maximum value of 176, which is the fleet size of the current bus network. Constraint (9) specifies the minimum allowable frequency, which is 4.8 buses per hour. To avoid too many stops on each route, the number of stops (including the terminal) for a route within TSW is limited to 9, which is specified by Constraint (10).

The value of T_{ie}, i.e., the average travel time from zone *i* to destination *e*, is a complex function of the route structure and frequency, which is defined by Constraint (11). The calculation of T_{ie} is based on the assumption of demand assignment made by Baaj and Mahmassani (1990). It is assumed that passengers will choose (a) the path that has the least possible number of transfers; and (b) the first bus arriving among a set of routes. Based on this assumption, the T_{ie}, value is determined in the following way. If there are direct bus routes from stop *i* to destination *e*, then the T_{ie} value will be the average travel time of all of these direct routes going from stop *i* to destination *e*, weighted by their frequencies, plus the expected waiting time for a bus of these routes. If there is no direct route at stop *i*, then the T_{ie} value will be calculated as the average travel time of all of the indirect routes stopping at stop *i*, weighted by their frequencies, plus the average waiting time for these routes. The T_{ie}^{n}, i.e., travel time for the *n*th route going from zone *i* to destination *e*, is calculated by Constraint (12). Constraint (13) calculates the integer variable RT_{ie}^{n}, which is l if the *n*th route goes from node *i* to destination *e*, and 0 otherwise. Constraint (14) calculates the integer variable NR_{ie}, which is l if there is no direct line from node *i* to destination *e*, and 0 otherwise.

It can be seen that constraints (8), (11), and (14) are all nonlinear, making the model a mixed-integer nonlinear programming one, which is *NP*-hard. To solve the investigated problem, a heuristic solution method is needed.

3. THE SOLUTION METHOD

In the proposed solution method, a specific genetic algorithm (GA) is developed to solve the route design problem, while a neighborhood search heuristic is integrated into the GA to solve the frequency setting problem. The frequency setting heuristic is implemented in the solution evaluation process of the GA. This will be discussed in greater detail later.

3.1 The genetic algorithm

Figure 2 illustrates the representation scheme of the genetic algorithm. The chromosome consists of 100 genes representing 10 routes. The first 10 genes represent the first bus route, which starts from node 1, goes through nodes 18, 15, 10, 12, and 7 and the TLT interchange stop (which is implicitly coded), and terminates at destination 25.

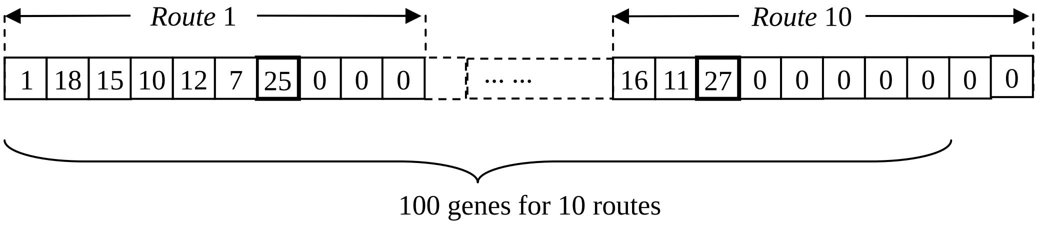


Figure 2. Representation scheme of the genetic algorithm.

Figure 3 illustrates the scheme of the GA. First, initial solutions are randomly generated. Then, the fitness values of the initial solutions are evaluated. It should be noted that the frequency setting heuristic is performed in the fitness evaluation process. Next, some parents are selected using the fitness-proportional roulette-wheel selection method. A certain number of offspring are generated through genetic operators. To improve the efficiency, a specific heuristic is applied to the offspring to improve the stop sequence of each route. The individuals of the new generation are selected randomly from both the parents and offspring. The algorithm is repeated until a certain number of generations have been generated.

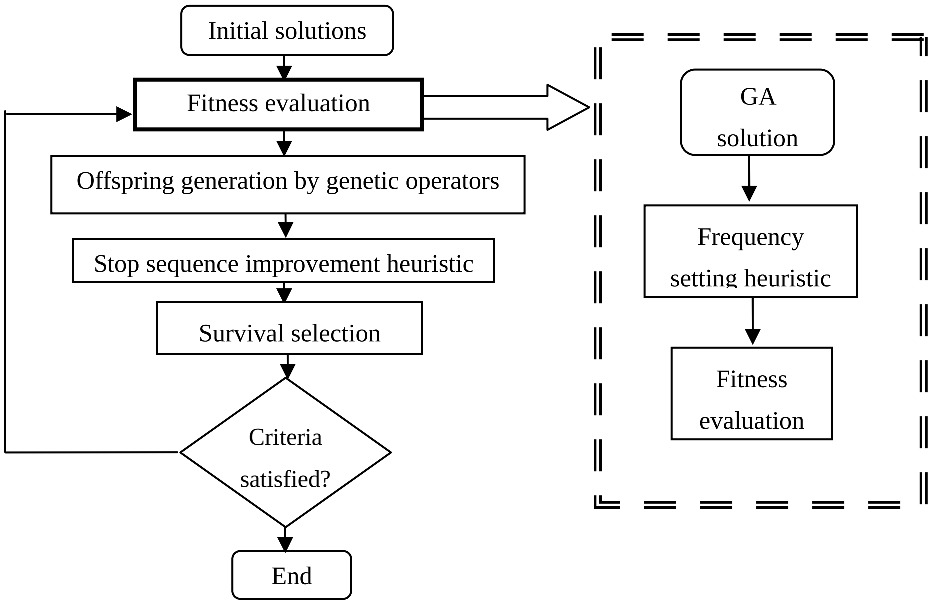
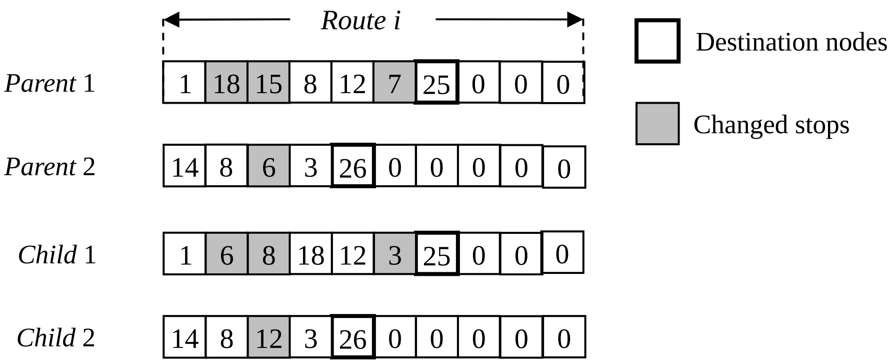


Figure 3. Detailed scheme of the genetic algorithm.

Two crossover operators and four mutation operators are developed, i.e., route crossover, stop crossover, insert mutation, remove mutation, swap mutation, and transfer mutation. The route crossover operator is designed to exchange routes between two solutions. It is similar to the two-point crossover. Rather than exchanging individual genes, the route crossover operator exchanges whole routes between two parents. The stop crossover operator is designed to exchange stops between two routes of two parents; i.e., intermediate stops in the corresponding routes of two parents are exchanged randomly to produce two children. Figure 4 illustrates the stop crossover process. First, the genes in the first child are copied from the first parent. Then, each of the intermediate stops in this child has a certain probability (20%) of being changed to a stop in the corresponding route of the second parent, while avoiding the same stops in each route. The second child is generated in a similar way with the parent roles reversed.



Parent 1	1	18	15	8	12	7	25	0	0	0
Parent 2	14	8	6	3	26	0	0	0	0	0
Child 1	1	6	8	18	12	3	25	0	0	0
Child 2	14	8	12	3	26	0	0	0	0	0

Figure 4. Illustration of the stop crossover process.

Four types of mutation operators are proposed for the GA, i.e., insert mutation, remove mutation, swap mutation, and transfer mutation. The insert mutation inserts a randomly selected stop into a randomly selected route in the solution. In contrast, the remove mutation randomly removes a stop from a route in the solution. The swap mutation exchanges two nodes of the same type (in terms of terminals, stops, and destinations) between two different routes of the solution. The transfer mutation randomly moves an intermediate stop in one route to another in the solution. For all of the mutation operators, a mechanism is adopted to prevent the same nodes from appearing on the same route.

3.2 Frequency setting heuristic

The fitness of the GA solution is set to be the reciprocal of the objective value. The frequency setting heuristic is adopted whenever the fitness of a GA solution is evaluated. Instead of explicitly coding the frequencies of these routes, the solution for the frequency setting heuristic is defined as a collection of the numbers of vehicles allocated to each route. The fleet size (total number of vehicles) constraint can be ensured by keeping the number of vehicles represented by the solution. The simplicity of this representation helps prevent infeasible solutions. A decent search heuristic is adopted, which is described as follows.

```
Step 1: Initialize a solution by randomly allocating vehicles to $N$ routes;
Step 2: For $i$ =1 to $N$
            For $j = i$ to $N$
                Move 1 vehicle from route $i$ to route $j$
                If the objective is improved, then go to Step 2, otherwise, undo the change
                Move 1 vehicle from route $j$ to route $i$
                If the objective is improved, then go to Step 2, otherwise, undo the change
            Next $j$
        Next $i$
Step 3: Output the best solution found
```

To ensure the minimum frequency of each route, a special treatment is adopted when calculating the objective value. If the frequency of a route is smaller than the minimum value, then a penalty is introduced by dividing the frequency by a nonnegative parameter.

3.3 Stop sequence improvement heuristic

Due to the great complexity of the problem, a decent search heuristic is adopted to improve the sequence of the stops of each route for every solution in the GA. To insure the efficiency, the objective of this heuristic is to minimize the trip time of the route, which does not depend on the frequency and thus is relatively easy to obtain.

4. EXPERIMENTS

Computational experiments were conducted for fine-tuning and verification of the proposed solution method. The solution method was coded in Visual C++ 2003, and run on a 1.73 GHz computer with 1 GB RAM.

4.1 Fine-tuning of the GA

Experiments were conducted to fine-tune the GA. Eight replicates of the GA with different settings were implemented. Each GA was run 20 times. Table 1 summarizes the computational results. For all eight GAs, the population size is 20. In each generation, 16 offspring are generated. Then, 20 individuals for the new generation are randomly selected from both the parents and offspring. The weighting parameters for the transfers and total travel time, i.e., B_1 and B_2, are set to be 30 and 1, respectively. Each run continues until 500 generations are produced. It should be noted that the preliminary results showed that the stop sequence improvement heuristic proposed in Section 3.3 improved the objective values by 5-8%. Therefore, the stop sequence improvement heuristic was included in all of the experiments.

For all eight GAs (GA-1 to GA-8), the computational time is similar and acceptable (2 to 3 minutes). GA-1 uses only route crossover to generate offspring, while GA-2 uses only stop crossover. It can be seen that GA-1 and GA-2 have similar performance in terms of fitness value. GA-3 uses both crossover operators; i.e., for each pair of selected parents, either route crossover or stop crossover is randomly selected with the same possibility. GA-3 achieves a better fitness value than either of the previous two GAs, indicating that it is advantageous to combine the two crossover operators. GA-4, GA-5, GA-6, and GA-7 are the same as GA-3 except that after using crossover operators, they also adopt the insert, delete, swap, and transfer mutation operators, respectively. All four GAs have better performance than GA-3. In GA-8, all four mutation operators are combined in the algorithm; i.e., for each child randomly generated by either route or stop crossover operators, one of the four mutation operators is randomly applied. The selection possibilities of the four mutation operators are 0.4, 0.4, 0.1, and 0.1, respectively. The results show that GA-8 obtained the best fitness value among all GAs, indicating that it is also advantageous to combine different mutation operators in the algorithm. In the remaining experiments, the operator setting and relative parameters in GA-8 are used.

Table 1. Computational results of different GA settings

	GA-1	GA-2	GA-3	GA-4	GA-5	GA-6	GA-7	GA-8
fitness(10^{-7})[a]	5.81	5.85	5.92	6.00	5.97	6.08	6.13	6.22
std. dev.[b]	0.03	0.03	0.04	0.05	0.04	0.04	0.01	0.02
transfers[a]	10640	10246	9230	6922	10890	8695	8327	7064
std. dev.[b]	864	1575	808	633	906	822	374	580
travel time[a]	1401730	1403381	1411921	1458833	1348236	1385306	1380719	1396851
std. dev.[b]	27949	51237	28222	28666	30327	29092	12459	20226
CPU time[a]	2.48	2.53	2.47	2.48	2.54	2.43	2.35	2.50
std. dev.[b]	0.05	0.05	0.04	0.04	0.06	0.04	0.05	0.03

[a] Average value for 20 runs.
[b] Standard deviation.

Table 2. Best solution obtained by the proposed method

Routes	Stop sequence	Number of vehicles	Headway (minutes)
1	1, 5, 9, 7, 8, 16, 17, T, 24	19	6.4
2	7, 6, 1, 5, 4, 12, 13, T, 25	19	8.6
3	20, 19, 23, 22, T, 24	12	7.9
4	9, 6, 7, 8, 15, 14, 19, T, 25	13	11.8
5	1, 5, 6, 7, 8, 16, 18, 23, T, 26	21	8.9
6	9, 7, 6, 1, 2, 3, 11, 12, 13, T, 24	23	5.9
7	14, 15, 10, 12, 18, 23, T, 28	17	6.9
8	7, 9, 10, 15, 14, 18, 23, 21, T, 24	15	8.2
9	7, 9, 10, 16, 18, 23, T, 25	16	9.1
10	1, 5, 6, 7, 9, 8, 16, T, 28	21	6.1

4.2 Comparison with the current network

There are 10 bus routes in the current network. The number of transfers is 6966 and the total travel time is 1480102 minutes. The solution for the comparison was obtained with a population size of 20 and 2000 generations. The algorithm was run 20 times. For all of the runs, both the number of transfers and total travel time are smaller than those for the existing network. On average, the number of transfers is reduced by 17.1%, while the total travel time is reduced by 7.11%. Table 2 shows the routes of the best solution among the 20 runs. The number of transfers is 5529, which is 20.6% smaller than that of the current network, and the total travel time is 1376336 minutes, which is 7.0% smaller than that of the current network. This is equivalent to 4.73 minutes less travel time for every user.

5 CONCLUSIONS

A solution to the bus route network design problem for a suburban residential area in Hong Kong has been investigated. The problem aims to reduce the number of transfers and total travel time of the network. Because these two objectives conflict and the latter is determined by both the route design and frequency setting, the route design and frequency setting problems are considered simultaneously. The proposed solution method integrates a specific genetic algorithm to optimize the route design and a neighborhood search heuristic to optimize the frequency setting. A new representation scheme is proposed in which the GA can search within all possible route structures. Two crossover and four mutation operators are developed. Experiments show that combining all of these operators improves the performance of the algorithm. The algorithm with reasonable parameter settings is also run multiple times. Compared to the current network, both the number of transfers and total travel time are reduced for all runs. In the best solution, the number of transfers is reduced by 20.6% and the total travel time by 7.0%.

ACKNOWLEDGEMENTS

An earlier version of this paper was presented and published in the *Proceedings of the 14th HKSTS International Conference*. We would like to thank the HKSTS for permitting us to reproduce the material for the HKIE Civil Division Conference 2010. The research was supported by a grant (201001159008) from the University Research Committee of the University of Hong Kong.

REFERENCES

Baaj, H. and Mahmassani, H.S. (1990) TRUST: A Lisp program for the analysis of transit route configurations. *Transportation Research Record* 1283, 125–135.

Ceder, A. and Wilson, N.H.M. (1986) Bus network design. *Transportation Research Part B* 20, 331–344.

Fan, W. and Machemehl, R. (2006a) Optimal transit route network design problem with variable transit demand: Genetic algorithm approach. *Journal of Transportation Engineering* 132(1), 40–51.

Fan, W. and Machemehl, R. (2006b) Using a simulated annealing algorithm to solve the transit route network design problem. *Journal of Transportation Engineering* 132(2), 122-132.

Guihaire, V. and Hao, J.K. (2008) Transit network design and scheduling: A global review. *Transportation Research Part A* 42, 1251–1273.

Lampkin, W. and Saalmans, P.D. (1967) The design of routes, service frequencies, and schedules for a municipal bus undertaking: A case study. *Operational Research Quarterly* 18(4), 375–397.

Tom, V.M. and Mohan, S. (2003) Transit route network design using frequency coded genetic algorithm. *Journal of Transportation Engineering* 129(2), 186–195.