Combined Simulated Annealing and Genetic Algorithm Approach to Bus Network Design

Conference Paper *in* Communications in Computer and Information Science · October 2010

DOI: 10.1007/978-3-642-16472-9_37 · Source: DBLP

Piotr S Olszewski
Warsaw University of Technology

Combined Simulated Annealing and Genetic Algorithm Approach to Bus Network Design

Li Liu[1], Piotr Olszewski[2], and Pong-Chai Goh[3]

[1] Shanghai Tat Hong Equipment Co., Beijing, PR China
lydia.liuli@gmail.com
[2] Warsaw University of Technology, Warsaw, Poland
p.olszewski@il.pw.edu.pl
[3] Nanyang Technological University, Singapore
cpcgoh@ntu.edu.sg

Abstract. A new method – combined simulated annealing (SA) and genetic algorithm (GA) approach is proposed to solve the problem of bus route design and frequency setting for a given road network with fixed bus stop locations and fixed travel demand. The method involves two steps: a set of candidate routes is generated first and then the best subset of these routes is selected by the combined SA and GA procedure. SA is the main process to search for a better solution to minimize the total system cost, comprising user and operator costs. GA is used as a sub-process to generate new solutions. Bus demand assignment on two alternative paths is performed at the solution evaluation stage. The method was implemented on four theoretical grid networks of different size and a benchmark network. Several GA operators (crossover and mutation) were utilized and tested for their effectiveness. The results show that the proposed method can efficiently converge to the optimal solution on a small network but computation time increases significantly with network size. The method can also be used for other transport operation management problems.

Keywords: Bus network design, optimization, genetic algorithm, simulated annealing.

1 Introduction

Improving the efficiency of public transport is an often-stated goal of transportation policy in big cities because only efficient public transport can successfully compete with private cars and thus help to ease the increasing traffic congestion. As buses are the backbone of public transport systems, optimization of bus routes would certainly contribute to improving the system efficiency. However, this problem is seldom tackled by transport planners. In most cases bus networks evolve incrementally – new services are being added as the city develops. In practice, bus route planning is often carried out by a combination of cognitive methods and trial-and-error.

In theory, optimization of bus routes for a city with a given road network, bus stop locations and passenger demand would involve deciding on the best number of routes and the best stop sequence for each route, with the objective of minimizing the sum of

J. Mikulski (Ed.): TST 2010, CCIS 104, pp. 335–346, 2010.

total passenger user costs and bus operating costs. It is basically a combinatorial problem and the number of possible solutions grows exponentially with the network size. Because of the problem complexity, literature on bus network optimization is relatively limited. Some early attempts to solve the problem analytically (e.g. [10]) involved greatly simplified networks and abstract models. The optimum values of key parameters could be determined by solving the objective function mathematically. However, real-life constraints on route length, route spacing, bus stop spacing and many-to-many character of demand cannot be reflected in these analytical models. This makes the analytical approach not practical for real life applications.

Another approach to solving the bus network design problem involves heuristic algorithms. Ceder and Wilson [3] considered both passenger and operator viewpoints and used a tree search algorithm to find all the feasible route sets. In practice, it is impossible for planners to examine all the solutions for a large network. It is helpful to use a screening algorithm, so as to find a limited number of better solutions before doing the final evaluation. Mandl [12] proposed an approach which started with a feasible set of routes and then searched for better solutions by trial and error. The improvement may be obtained by exchanging parts of routes, including a new node or excluding a node. Baaj and Mahmassani [2] proposed a similar hybrid route generation algorithm, starting with a simple skeleton network and extending it by adding new routes or inserting new nodes. The heuristic approaches have the advantage of using expert knowledge to reduce the search space.

With the increase of computer power, new approaches emerged for solving complex optimization problems with large search spaces. Genetic algorithm (GA) is one of the popular methods and has been applied to the bus route design problem [13]. Krishna Rao et al. [9] proposed a two-phase network design process using GA which was able to produce better results when compared with Mandl's study [12]. To validate the effectiveness of genetic algorithm, Chien et al. [4] applied the GA and exhaustive search to design a feeder bus route from an irregular service area with 160 zones. Identical optimal solutions were found by both methods but the GA was about 90 times faster. In order to extend the searching space, Lin et al. [11] suggested performing GA in parallel operations on three independent populations which were mixed and exchanged after each evolution. All of these studies proved that GA is efficient in searching for the optimum solution. However, it is known that GA may become stuck in a local optimum as it only accepts better solutions in its reproduction process. Yet accepting a worse solution is sometimes beneficial as it may lead to searching a wider space and finally reaching the global optimum.

Simulated annealing (SA) is also a promising algorithm with a strong search capability. However, its application in bus route design is very limited. Friesz et al. [6] applied SA to find an optimal network design when the flow pattern was constrained to be in equilibrium. In another study, a hybrid approach based on combing case-based reasoning and simulated annealing was proposed by Sadek [14]. Both of these studies showed that SA algorithm is efficient in solving optimization problems, although SA is not always capable of finding the global optimal solution. SA can sometimes accept worse solutions when certain requirements are met so as to extend its searching space beyond the local optimum.

It is natural to think that the combination of two powerful search algorithms may achieve better results, especially when one algorithm complements the other. Hence, a combined SA and GA approach is proposed in this paper. Studies on combined SA and GA are very rare. Zhao and Zeng [17] [18] used this method to search for an optimal solution for bus route network design problem. In their first study [17], the aim was to minimise the number of vehicle boardings that passengers have to make, while constraining the total route length and the number of routes. In their second study [18], Zhao and Zeng extended their method to optimise public transport network layout and headways with the objective of minimising total users' cost. They tested their method on Mandl's network and their solution was better than in previous studies. However, the objective function used by them is not quite practical as it ignores the bus system operating cost.

2 Problem Formulation

Bus route design involves not only planning of bus routes based on a road network, but also passenger demand assignment and frequency determination. Usually, the inputs to bus route design should comprise road network suitable for bus travel, bus stops locations and the stop-to-stop travel demand matrix. The aim of bus route design is to plan sufficient bus routes to cover all the bus stops and accommodate all the travel demand while not violating any constraints. Most common constraints are: fleet size, minimum and maximum service frequency etc. It is important to select a good objective function so that these constraints can be balanced. Based on a study by van Nes and Bovy [15], minimizing the total cost (the sum of operating cost and travelers' cost) is considered the desirable objective in bus network design with a fixed demand. This objective is also adopted in this study, i.e. the aim is to determine a set of bus routes that produces a minimum total cost while meeting all the requirements and constraints.

The bus route design problem can be formally presented in the following way. A road network $N = (Rd, B)$, comprises a set of roads $Rd = (1, 2 \ldots m)$ and a set of bus stops $B = (1, 2 \ldots n)$. A bus route is represented by a sequence of bus stop IDs. A set of candidate bus routes $R \{1, 2 \ldots k\}$ is generated for network N. Each bus route is given a unique number as its ID. A solution which is a subset of candidate routes, $R_{SR} \{1, 2 \ldots\}$, is represented by a sequence of bus route IDs. For example, a route network consisting of 5 bus routes: 4, 27, 34, 9, and 12, will be simply represented by a string with the length of 5 elements.

The objective function is to minimize the total bus system cost. This is expressed as follows (adopted from [13]):

$$Min : E = c_p \sum_{i=1}^{n} \sum_{j=1}^{n} T_{ij} \cdot t_{ij} + c_b \sum_{k \in R_{SR}} f_k L_k \ . \tag{1}$$

subject to the following constraints:

$f_k \geq f_{\min} \forall k \in R_{SR}$ (frequency feasibility)

$l_k \leq l_{\max} \forall k \in R_{SR}$ (load factor constraint)

where:

c_b unit bus operating cost per kilometer ($/bus-km)

c_p unit cost of passenger travel time ($/pass.-h)

E	the value of objective function ($/h)
f_k	frequency of k^{th} bus route (buses/h)
f_{min}	minimum bus operating frequency (buses/h)
k	bus route number
l_k	load factor on k^{th} bus route = $Q_k^{\max} /(f_k \cdot bus\ capacity)$
l_{max}	maximum allowable load factor of bus service
L_k	round trip distance for k^{th} bus route (km)
n	number of bus stops in the network
T_{ij}	travel demand from node i to j (pass./h)
t_{ij}	total travel time (sum of in-vehicle time, waiting time and transfer time) between stops i and j (h)

The total travel time between bus stops i and j (t_{ij}) is the sum of in-vehicle travel time, waiting time and transfer time. The first term in Eq. 1 represents the total travel cost incurred by the traveler and the second term represents the total bus operating cost of all the routes in the solution route set. Hence, the sum of both terms represents the total system cost. This model will be solved using the proposed SA-GA approach.

3 Framework of the Proposed Method

The proposed hybrid SA-GA method is used to design bus routes. SA is used as the main algorithm to search for the optimal solution. In each SA iteration, GA is used to give a random disturbance to the current solution. Since every solution is actually a set of bus routes, it is necessary to generate enough candidate routes for each pair of bus stops in a road network before running SA-GA. This is done with the k-shortest path algorithm [8].

The main structure of SA-GA process is presented in Fig. 1. It is implemented in Visual C++ environment. The initial network N_0 is generated by a GA sub-function called *initialization* and its objective function value, E_0, is calculated. The maximum temperature (T) is set to 50. Parameter I is used to control the iterations of the GA process. The maximum number of iterations is set to 20. A new solution N_1 is obtained from the GA process and evaluated to obtain E_1. If the new solution is better than the old one (i.e. $E_1 < E_0$), it is accepted and N_1 replaces N_0. Even if the new solution is worse than N_0 (i.e. $E_1 - E_0 > 0$), it can still be accepted randomly, with the acceptance probability which is a function of the two objective function values and the current temperature (Fig 1).

GA is used to generate new potential solutions among which the one with the best fitness value is selected and subsequently used by the SA process for further selection [7]. In the case of minimization, the fitness value is usually assumed to be the reciprocal of the objective function value.

During the initialization phase, two tasks are accomplished. The first task is to generate a solution (a string of bus route IDs) N_0 which will be used as the initial solution. Since the size of the optimum solution is not known beforehand, variable string length coding is applied. The size of the solution *(s)* is selected at random within a predefined range and solution N_0 is generated by selecting s candidate routes at random.

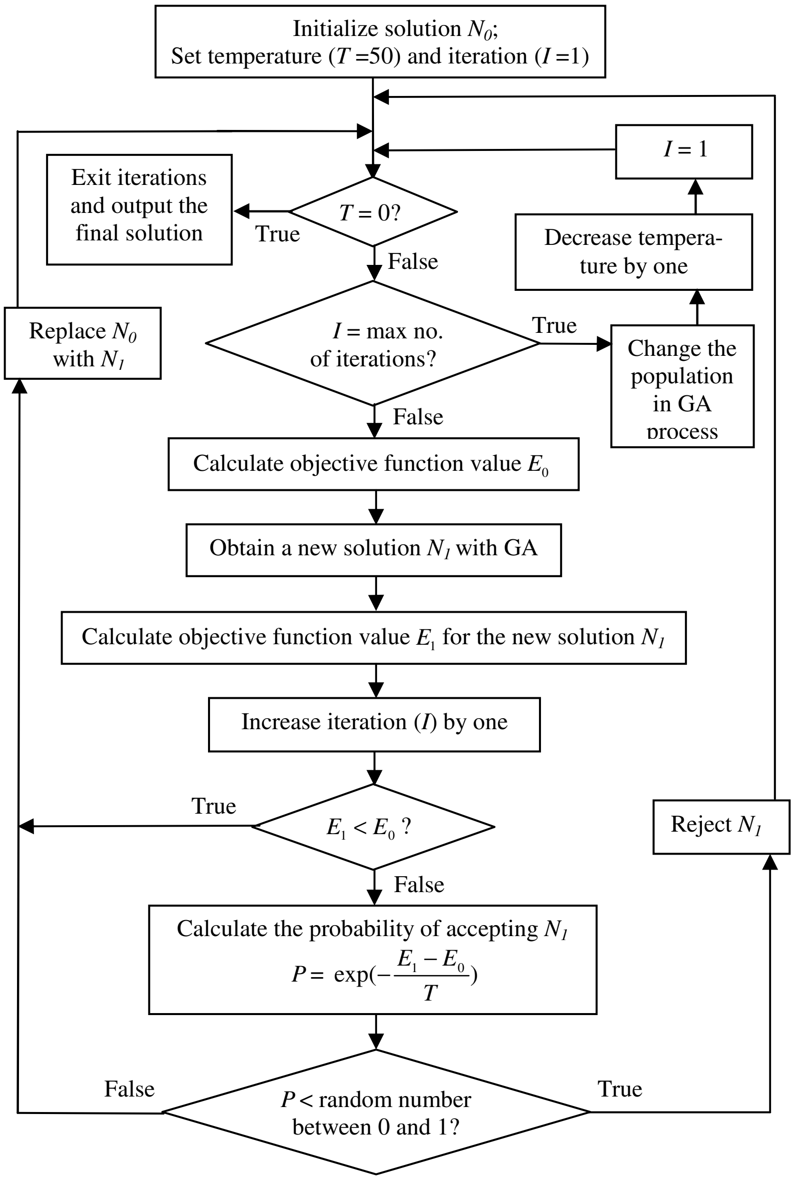

Fig. 1. Flowchart of the proposed SA-GA main process

The second task is to generate a population of solutions. Bigger population size gives better diversity and hence the chance of finding better solutions. However, bigger population increases the computation time. In this study, population size of 200 was chosen after the initial experiments showed that this did not increase the computation time dramatically. Using the same method as for initializing $N_{0,}$ the other 200 solutions of the population are generated. Each individual solution is evaluated based on the objective function formula and its fitness value is determined for reproduction purpose in the GA process.

Reproduction is a process in which some strings with better fitness values are selected from the population for further GA operations. Before doing reproduction, N_0 is combined with the 199 strings to form a population with 200 strings. The popular roulette wheel selection method [7] is used in this study. In this way, strings with a higher fitness value have a higher probability of being selected.

Crossover is a process to generate new strings from a given pair of strings in the process of reproduction. For a set of strings, generally crossover is performed between the first and second strings, third and fourth, and so on. In addition, performing the crossover on a pair of strings is controlled by crossover probability P_c, which has a predefined value. To maximize the exploration space, P_c is set to 1 and crossover is performed on every pair of strings. Several crossover methods are available. Each has its characteristics and is potentially more efficient for certain types of problems. The common crossover approach applied in most GA applications is single-point crossover [7]. To test the effectiveness of different methods, three other kinds of crossover: two-point, uniform and convex are also used in this study.

Mutation is the occasional random alteration of the value of an element of a string. The mutation operator helps to introduce some potentially useful components (i.e. good bus stop sequences). As in crossover, this process is also controlled by a probability parameter P_m, (mutation probability). The frequency of mutation is usually very small in GA applications. One mutation per thousand elements is a typical value and produced best results in most applications. The value of $P_m = 0.001$ is also used in this study. Two types of mutations are used: uniform and dynamic. It should be noted that each type of mutation may result in duplicate elements in child strings. Since in the bus route network design problem no duplication is allowed, an additional procedure to remove duplicates is performed after a mutation. In this process, each duplicate is replaced by a route ID chosen randomly from candidate routes.

4 Application to Theoretical Networks

The SA-GA approach is applied to four theoretical networks to test its performance. The first network (N1) is shown in Fig. 2 and represents a two by three square grid with each road link 600 meters long. There are 14 bus stops (large dots) distributed on this network. Three of the bus stops are designated as terminals (crosses). These are the end points of all bus routes. Location of all bus stops is assumed fixed. The demand between all pairs of bus stops is also fixed. All bus stops, bus terminals and road intersections are considered nodes. Links between any pair of nodes are called bus links. It is assumed that all roads are two-directional. Three bigger theoretical networks (N2, N3 and N4) are generated in a similar way. Their characteristics are summarized in Table 1.

The task of candidate route generation should ideally be done with the k-shortest simple paths algorithm [8]. However, this algorithm has a high complexity and its computation time would be extremely large. Therefore, a modified Dijkstra algorithm [5] is applied to find the candidate routes between all pairs of bus terminals in order to save computation time. An amendment was made to this algorithm to exclude all paths with closed loops, as in reality a closed loop would normally not be acceptable as part of a bus route. It is not practical to ensure within an acceptable computation time limit that k candidate routes generated are the k shortest paths. However, the method guarantees that the shortest path is included as one of the candidate routes. The numbers of candidate routes generated for all four theoretical networks and the Mandl network are shown in Table 1. The process of route generation for all the networks is done within 30 seconds.

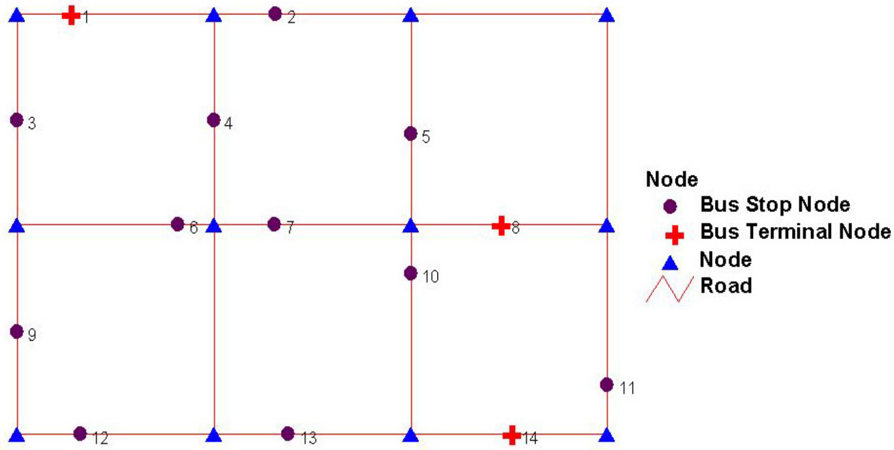

Fig. 2. Test network N1

Table 1. Characteristics of test networks

Network	No. of bus stops	No. of terminals	No. of nodes	No. of bus links	No. of candidate routes
N1	14	3	11	29	21
N2	20	4	14	40	171
N3	26	5	19	54	454
N4	32	6	23	67	720
Mandl	15	15	15	21	1183

To make the application more realistic, two-path assignment is used in this study. Two paths are considered between any two bus stops and used in travel demand assignment. Therefore, the objective function (Eq. 1) is modified as follows:

$$Min : E = c_p \sum_{i=1}^{n} \sum_{j=1}^{n} (T_{ij}^1 \cdot t_{ij}^1 + T_{ij}^2 \cdot t_{ij}^2 + T_{ij} \cdot t_{wij}) + c_b \sum_{k \in R_{SR}} f_k \cdot L_k \, z\,. \quad (2)$$

subject to the following constraints:

i. all bus stops are covered

ii. $f_k \geq f_{\min}$

where:

T_{ij}^1 travel demand from stop i to j assigned to the first path (pass./h)

T_{ij}^2 travel demand from stop i to j assigned to the second path (pass./h)

t_{ij}^1 total travel time for the first path between stops i and j (h)

t_{ij}^2 total travel time for the second path found between stops i and j (h)

t_{wij} average waiting time for travel form stop i to j (h)

The superscripts 1 and 2 represent the two paths used in the assignment – the first path is always the shortest and the second is an alternative path between stops i and j. The waiting time for passengers traveling from bus stop i to j (t_{wij}), is calculated as half of the headway determined considering buses forming the paths from i to j.

Usually, a frequency constraint is also imposed for the objective function. For a bus route with a frequency lower than the minimum frequency, the system will check whether the bus stop coverage constraint will be violated if this route was excluded from the solution. If the constraint is not violated, this route will be excluded and the first iteration of demand assignment is performed again. Otherwise this route will be kept and the frequency for this route is set to the predefined minimum frequency. However, a maximum frequency constraint was not used as this may exclude some very good routes which carry a lot of travelers. In real life, this problem can be solved manually by splitting one heavily loaded route into two similar routes.

Before performing the SA-GA, some parameter values need to be specified:

- Bus capacity: 80 persons/bus;
- Average bus speed: 18 km/h;
- Travelers' value of time c_p: \$8.00/hour [16];
- Bus operating cost c_b: \$5.24/bus-km (based on US public transport data [1]);
- Transfer penalty: 400 sec/transfer, assumed based on preliminary results.

Evaluating the solution is probably the most important and complicated part of the whole process. Since the solution is randomly generated and always changing, two sets of values need to be found before calculation of the objective function value and fitness based on Eq. 2. These are the in-vehicle travel times between all pairs of bus stops and frequencies for all bus routes in the solution. Frequency is obtained by dividing the maximum demand for each route by bus capacity. The demand for each route will keep changing while different routes are selected into the solution network. Therefore, it is necessary to do travel demand assignment every time a solution is formed or changed. The method adopted is to assign demand to the two selected paths. The rules for selecting the two paths are:

- the first path must be the shortest path between the pair of bus stops; and
- the in-vehicle travel time of the second path can not be more than 20% longer than the in-vehicle travel time of the shortest path.

Transit assignment is done in two iterations. In the first iteration, the demand between each pair of bus stops (T_{ij}) is assigned equally to the two selected routes and the frequencies for both routes are calculated. In the second iteration, the demand is re-assigned to the two routes according to their calculated frequencies.

Two options are considered when transfers have to be involved. One is direct transfer at the alighting bus stop. The other is an inter-connecting transfer by walking from the alighting bus stop to a different bus stop to transfer to another service. The inter-connecting limit is set to 300 meters of walking distance. The transfer penalty and walking time are added to the total path time.

The final frequency of each route can be calculated based on the resulting demand from the second transit assignment. The average waiting time for each trip can also be calculated. Now all the data needed for evaluation based on Eq. 2 are available.

The SA-GA approach as described above was first applied to network N1. Since four types of crossover and two types of mutation were used, in total eight different combinations were tested. The average computation time for one combination was 7 minutes for network N1. The overall best solution for network N1 is presented in Fig. 3. With this solution, only 38 stop-to-stop trips out of 182 possible combinations need one transfer in the shortest travel path. The others are all direct paths without transfer. The actual transfer rate is 7589/6434 = 1.18 transfers per trip. This means that 85% of trips were accomplished without any transfer.

Table 2. Summary of computation results of all networks

Network	Crossover used by the best result	Mutation used by the best result	Objective function value ($/h)	Solution size (routes)	Average time for SA-GA (min)
N1	Single/Two point/Uniform	Uniform/ Dynamic	10030	4	7
N2	Two-point	Dynamic	19986	6	37
N3	Single-point	Dynamic	36599	10	81
N4	Two-point	Dynamic	61698	19	133
Mandl (Min. time)	Two-point	Dynamic	26765	10	35
Mandl (Min. cost)	Two-point	Dynamic	24465	6	35

The best results for the other three bigger networks are presented in Table 2. With larger network size, the computation time has also increased significantly. The same procedure was applied to the benchmark network which was first presented by Mandl [12] and then used by many other researchers [2] [18]**.** The characteristics of Mandl's network were given in Table 1. To make the results comparable with the previous studies, two objective functions were used: minimising the total travel time and minimising the total cost.

The solution from SA-GA with minimisation of travel time as the objective, produced the result 37.8% better than Mandl's final solution and 22.6% better than Baaj and Mahmassani's best solution in terms of percentage of direct trips. In terms of total travel time, the result obtained is 17% better than Mandl's solution; 12% better than Baaj and Mahmassani's solution and 3% better than Zhao and Zeng's solution.

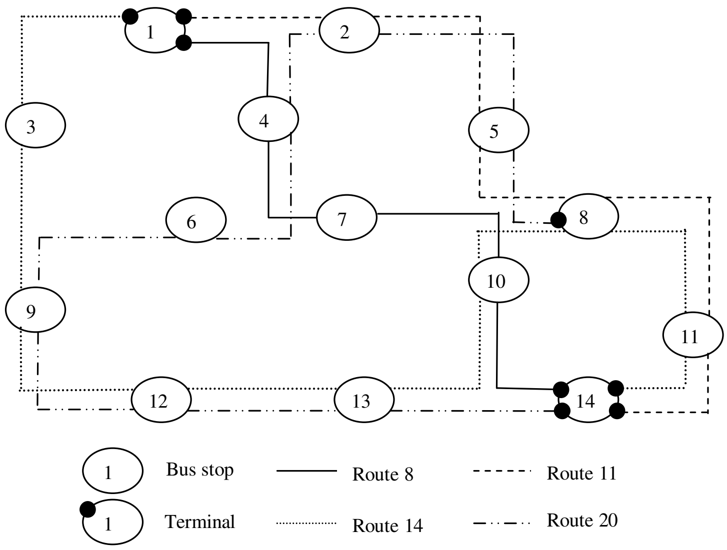

Fig. 3. Best solution for network N1

5 Discussion

The computation times of all the networks are indicated in Table 2. The computation time goes up very quickly with the increase of network size. This is not surprising because bus network design problem is a highly computationally-intensive problem (NP-hard). After examining the results of network N1, it seems very likely that an optimal or nearly optimal solution is found because the same solution is obtained in 5 combinations of mutation and crossover methods out of a total of 8. When comparing the features of the best results for all four networks, it can be seen that all the best solutions came from dynamic mutation and three best solutions came from the two-point crossover. This suggests that dynamic mutation and two-point crossover are more suitable than the other mutation and crossover techniques for solving the bus network design problems with GA.

Some of the parameters used for SA calculations like the starting temperature and number of iterations conducted at each temperature were selected in an attempt to balance the computation time and the size of search space. The selection of parameters of GA operations like population size, probability to conduct crossover (P_c) and probability to perform mutation (P_m) may affect the performance of SA-GA approach.

Since there is no scientific method for choosing the "optimum value" for these parameters, sensitivity tests were run to test the effectiveness of these parameter values. The best values obtained from sensitivity tests were used in the applications of SA-GA to both theoretical and the benchmark networks.

On the other hand, selection of some of the other parameters was tentative and may require further testing. For example, transit assignment was done in two iterations and only two paths were used. In the first iteration, demand was assigned equally to the two paths and in the second, assignment was based on bus frequency. The number of paths considered, the acceptable time range and the number of iterations were assumed in this way in order to control the computation time. Whether using three or more paths, setting larger acceptable time range or more assignment iterations would produce better results should be the subject of further investigation.

6 Conclusions

The bus route design problem is well known for its complexity and computational intensity as the number of possible combinations increases exponentially with network size. This paper has presented a combined SA-GA approach for bus route network design. It is implemented in two phases: a set of candidate routes is first developed and then an optimal subset of routes is selected. With the help of advanced algorithms: simulated annealing (SA) and genetic algorithm (GA), the design problem is solved in minutes. The SA-GA solution of Mandl's network is better than previous solutions and the in-vehicle travel time is quite near to its theoretical minimum. Moreover, a more realistic objective function and evaluation method are applied: demand is assigned to two paths for every pair of bus stops using an iterative procedure. This method has not been used in solving the bus route design problem before.

Results of the method application to four theoretical networks suggest that different genetic operators have different levels of efficiency. Since all the best solutions come from dynamic mutation and three best solutions come from two-point crossover, the indication is that dynamic mutation and two-point crossover are the most suitable for solving the bus network design problem with GA.

In terms of computation time, it is expected that the time will increase quickly and is approximately proportional to the cube of the size of the network. This result is not unexpected, as the bus route design problem has been proven to be one of the NP-hard problems which are nearly impossible to solve in polynomial time.

While the effectiveness of the proposed SA-GA approach has been demonstrated using theoretical networks, it is intended to further explore its effectiveness on a real bus network. In a case study, part of Singapore road network with around 400 links, 80 bus stops and 8 bus terminals will be used for bus route planning with SA-GA.

The proposed method can also be used for other transport operation management problems, for example: routing of demand-responsive transport, planning temporary bus routes for special events or cases when part of the network is closed due to maintenance or construction activities.

References

1. APTA: 2009 Public Transportation Fact Book. American Public Transit Association, http://www.apta.com/resources/statistics/Pages/transitstats.aspx
2. Baaj, M.H., Mahmassani, H.S.: Hybrid Route Generation Heuristic Algorithm for the Design of Transit Networks. Transportation Research Part C 3, 31–50 (1995)
3. Ceder, A., Wilson, N.H.M.: Bus Network Design. Transportation Research Part B 20, 331–344 (1986)
4. Chien, S., Yang, Z., Hou, E.: Genetic Algorithm Approach for Transit Route Planning and Design. J. of Transportation Engineering 127, 200–207 (2001)
5. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. The MIT Press, Cambridge (2001)
6. Friesz, T.L., et al.: A Simulated Annealing Approach to the Network Design Problem with Variational Inequality Constraints. Transportation Science 26, 18–26 (1992)
7. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Co., Reading (1989)
8. Hershberger, J., Maxel, M., Suri, S.: Finding the k Shortest Simple Paths: a New Algorithm and its Implementation. ALENEX, Baltimore (2003)
9. Krishna Rao, K.V., Muralidhar, S., Dhingra, S.L.: Public Transport Routing and Scheduling Using Genetic Algorithms. In: Proc. 3rd Int. Workshop on Transportation Planning and Implementation Methodologies for Developing Countries, IIT Bombay, pp. 91–102 (1998)
10. Kuah, G.K., Perl, J.: Optimization of Feeder Bus Routes and Bus-Stop Spacing. J. of Transportation Engineering 114, 341–354 (1988)
11. Lin, X.H., Kwok, Y.K., Lau, V.K.N.: A Genetic Algorithm Based Approach to Route Selection and Capacity Flow Assignment. Computer Communications 26, 950–960 (2003)
12. Mandl, C.E.: Evaluation and Optimisation of Urban Public Transport Networks. European J. of Operational Research 5, 396–404 (1980)
13. Pattnaik, S.B., Mohan, S., Tom, V.M.: Urban Bus Transit Route Network Design Using Genetic Algorithm. J. of Transportation Engineering 124, 368–375 (1998)
14. Sadek, A.W.: Hybrid Simulated Annealing and Case-Based Reasoning Approach for Computationally Intensive Transportation Problems. Transportation Research Record 1774, 18–24 (2001)
15. van Nes, R., Bovy, P.H.L.: The Importance of Objectives in Urban Transit Network Design. In: 79th Annual Meeting of the Transportation Research Board, Washington D.C. (2000)
16. VTPI: Transit Evaluation. Online TMD Encyclopedia, Victoria Transport Policy Institute, http://www.vtpi.org/tdm/tdm62.htm
17. Zhao, F., Zeng, X.: Optimization of Transit Network Layout and Headway with a Combined Genetic Algorithm and Simulated Annealing Method. Engineering Optimization 38, 701–722 (2006)
18. Zhao, F., Zeng, X.: Simulated Annealing-Genetic Algorithm for Transit Network Optimization. J. of Computing in Civil Engineering 20, 57–68 (2006)