A Tabu Search Based Heuristic Method for the Transit Route Network Design Problem

Wei Fan and Randy B. Machemehl

Department of Civil Engineering, Ernest Cockrell, Jr. Hall, 6.9, University of Texas at Austin, Austin, TX 78712-1076, USA
{weifan, rbm}@mail.utexas.edu

Summary. Systematic tabu search based meta-heuristic algorithms are designed and implemented for the transit route network design problem. A multi-objective nonlinear mixed integer model is formulated. Solution methodologies based on three variations of tabu search methods are proposed and tested using a small experimental network as a pilot study. Sensitivity analysis is performed, a comprehensive characteristics analysis is conducted and numerical results indicate that the preferred tabu search method outperforms the genetic algorithm used as a benchmark.

1 Introduction

Public transit has been widely recognized as a potential way of reducing air pollution, lowering energy consumption, improving mobility and lessening traffic congestion. Designing an operationally and economically efficient bus transit network is very important for the urban area's social, economic and physical structure.

Generally speaking, the network design problem involves the minimization (or maximization) of some intended objective subject to a variety of constraints, which reflect system performance requirements and/or resource limitations. In the past decade, several research efforts have examined the *bus transit route network design problem* (BTRNDP). Previous approaches that were used to solve the BTRNDP can be classified into three categories: 1) Practical guidelines and ad hoc procedures; 2) Analytical optimization models for idealized situations; and 3) Meta-heuristic approaches for more practical problems. NCHRP Synthesis of Highway Practice 69 (1980) provides industry rule-of-thumb service planning guidelines. Furthermore, in the early research efforts, traditional operations research analytical optimization models were used. Rather than determining both the route structure and design parameters simultaneously, these analytical optimization models were primarily applied to determine one or several design parameters (e.g., stop spacing, route spacing, route length, bus size and/or frequency of service) on a predetermined transit route network structure. Generally speaking, these models are very effective in solving optimization-related problems for networks of small size or with one or two decision variables. However, when it comes to the transit route design problem for a network of realistic size in which many parameters need to be determined, this approach does not work very well. Due to the inherent complexity involved in the BTRNDP, the meta-heuristic approaches, which pursue reasonably good local optima but do not guarantee finding the global optimal solution, were therefore proposed. The meta-heuristic approaches primarily dealt with simultaneous design of the transit route network and determination of its associated bus frequencies. Examples of the general heuristic approaches can be seen in the work of Ceder and Wilson (1986), Baaj and Mahmassani (1992), and Shih *et al.* (1998). Genetic algorithm-based heuristic approaches that were used to solve the BTRNDP can be seen in Pattnaik *et al.* (1998), Chien *et al.* (2001) and Fan and Machemehl (2004).

However, the major shortcoming of most previous approaches is that they did not study the BTRNDP in the context of the "distribution node" (or bus stop) level and simply aggregate zonal travel demand into a single node. This precludes them as generally accepted applications for practical transportation networks because the frequency-based rule for the traditional transit trip assignment model based on this assumption is incorrect. Therefore, the BTRNDP should be considered in a more general real world situation. Furthermore, previous research efforts mainly centered on genetic algorithms and other potential heuristic algorithms such as tabu search methods are seldom used to solve the BTRNDP. To search for possibly good and/or better network solutions, these methods should be considered.

The objective of this paper is to systematically examine the underlying characteristics of the optimal BTRNDP in the context of the "distribution node" level. A multi-objective nonlinear mixed integer model is formulated for the BTRNDP. Characteristics and model structures of the Tabu Search (TS) algorithms are reviewed. A TS algorithm-based solution methodology is proposed. Three different variations of TS algorithms are employed and compared as the solution method for finding an optimum set of routes from the huge solution space. A genetic algorithm is also used as a benchmark to measure the quality of the TS methods. Numerical results including sensitivity analysis and characteristics identification are presented using an experimental network. The subsequent sections of this paper are organized as follows. Section 2 presents the model formulation of the BTRNDP from a systematic view. The objective function and related constraints are also described. Section 3 discusses general characteristics of the TS algorithms. Section 4 proposes the solution methodology for the BTRNDP, which contains three main components: an initial candidate route set generation procedure; a network analysis procedure and a TS procedure that guides the candidate solution generation process. Section 5 presents the applications of the proposed solution methodology to an experimental network and the numerical results are also discussed. Finally in Section 6, a summary concludes this paper.

2 Model Formulation

Essentially speaking, the transportation system is described in terms of "nodes," "links" and "routes." A *node* is used to represent a specific point for loading, unloading and/or transfer in a transportation network. Generally speaking, there are three kinds of nodes in a bus transit network system: (a) Nodes representing centroids of specific zones; (b) Nodes representing road intersections; and (c) Nodes with which zone centroid nodes are connected to the network through centroid connectors. Note that nodes could be real (identifiable on the ground) or fictitious. Furthermore, the term "distribution nodes" is introduced especially for the third kind of node. A *link* joins a pair of nodes and represents a particular mode of transportation between these nodes, which means that if two modes of transportation are involved with the same link, these are represented as two links, say walk mode and transit mode. This is natural since the travel time associated with every mode-specific link is different. A *route* is a sequence of nodes. Every consecutive pair of the node sequence must be connected by a link of the relevant mode. The bus line headway on any particular route is the inter-arrival time of buses running on that route. A *graph* (network) refers to an entity G = $\{N, A\}$ consisting of a finite set of N nodes and a finite set of A links (arcs) which connect pairs of nodes. A *transfer path* is a progressive path that uses more than one route. Note that a typical geographical zone system may be based upon census boundaries and all land areas are encompassed by streets or major physical barriers. The zone centroids are located somewhere near the centers of the zones and zone connectors are used to connect these centroids to the modeled network. Generally, the centroid node represents the "demand" center (origin and/or destination) of a specific traffic zone. Distribution nodes are the junctions of centroid connectors and road links and might physically represent bus stops. It should be pointed out that centroid connectors are usually fictitious and they are used as the origins and/or destinations for implementation of the shortest path and k shortest path algorithms. Furthermore, an important characteristic of these centroid connectors is the distances that transit users have to walk to get to the routes that provide service to their intended destinations. Note that the terms, "arc" and "link" are used interchangeably.

Consider a connected network composed of a directed graph G = $\{N, A\}$ with a finite number of nodes and arcs. The following notations are used.

Sets/Indices

$i, j \in N$ centroid nodes (i.e., zones)
$r_k \in R$ routes
$i_t \in N$ t-th distribution node of centroid node i
$tr \in R$ transfer paths that use more than one route from R

Data

R_{max} maximum allowed number of routes for the route network

D_{max}	maximum length of any route in the transit network
D_{min}	minimum length of any route in the transit network
d_{ij}	bus transit travel demand between centroid nodes i and j
h_{max}	maximum headway required for any route; (say, 60 minutes)
h_{min}	minimum headway required for any route; (say, 5 minutes)
L_{max}	maximum load factor for any route
P	seating capacity of buses operating on the network
W	maximum bus fleet size available for operations on the route network
C_v	per-hour operating cost of a bus; ($/vehicle/hour)
C_m	value of time; ($/minute)
O_v	operating hours for the bus running on any route; (hours)
C_d	value of one unsatisfied transit demand in dollars; ($/person)
C_i	$(i = 1, 2, 3)$ weights reflecting the relative importance of three components including the user costs, operator costs and unsatisfied total demand costs, respectively; note that $C_1 + C_2 + C_3 = 1$

Decision Variables

M	the number of routes of the current proposed bus transit network solution
r_m	the m-th route of the proposed solution, $m = 1, 2, \ldots, M$
D_{r_m}	the overall length of route r_m
$d_{ij}^{r_m}$	the bus transit travel demand between centroid nodes i and j on route r_m
d_{ij}^{tr}	the bus transit travel demand between centroid nodes i and j along transfer path tr
DR_{ij}	the set of direct routes used to serve the demand from centroid nodes i and j
TR_{ij}	the set of transfer paths used to serve the demand from centroid nodes i and j
$t_{ij}^{r_m}$	the total travel time between centroid node i and j on route r_m
t_{ij}^{tr}	the total travel time between centroid node i and j along transfer path tr
h_{r_m}	the bus headway operating on route r_m; (minutes/vehicle)
L_{r_m}	loading factor in route r_m
T_{r_m}	the round trip time of route r_m; $T_{r_m} = 2D_{r_m}/V_b$
N_{r_m}	the number of operating buses required on route r_m; $N_{r_m} = T_{r_m}/h_{r_m}$
$Q_{r_m}^{max}$	the maximum flow occurring on the route r_m

Objective Function

The objective is to minimize the sum of operator cost, user cost and unsatisfied demand costs for the studied bus transit network. The objective function is as follows:

$$\min z = C_1 \cdot \left(\sum_{i \in N} \sum_{j \in N} \sum_{r_m \in DR_{ij}} d_{ij}^{r_m} t_{ij}^{r_m} + \sum_{i \in N} \sum_{j \in N} \sum_{tr \in TR_{ij}} d_{ij}^{tr} t_{ij}^{tr} \right) +$$
$$C_2 \cdot \frac{C_v}{C_m} \cdot O_v \cdot \left(\sum_{m=1}^{M} \frac{T_{r_m}}{h_{r_m}} \right) +$$
$$C_3 \cdot \frac{C_d}{C_m} \cdot \left(\sum_{i \in N} \sum_{j \in N} d_{ij} - \sum_{i \in N} \sum_{j \in N} \sum_{r_m \in DR_{ij}} d_{ij}^{r_m} - \sum_{i \in N} \sum_{j \in N} \sum_{tr \in TR_{ij}} d_{ij}^{tr} \right)$$

s.t.

$h_{min} \le h_{r_m} \le h_{max}$ $\quad r_m \in R$ (headway feasibility constraint)

$L_{r_m} = \frac{Q_{r_m}^{max} \cdot h_{r_m}}{P} \le L_{max}$ $\quad r_m \in R$ (load factor constraint)

$\sum_{m=1}^{M} N_{r_m} = \sum_{m=1}^{M} \frac{T_{r_m}}{h_{r_m}} \le W$ $\quad r_m \in R$ (fleet size constraint)

$D_{min} \le D_{r_m} \le D_{max}$ $\quad r_m \in R$ (trip length constraint)

$M \le R_{max}$ (maximum number of routes constraint)

$M, h_{r_m}, N_{r_m}, Q_{r_m}^{max}, d_{ij}^{r_m}, d_{ij}^{tr}$, are all integers.

The first term of the objective function is the total user cost (including the user cost on direct routes and that on transfer paths), the second part is the total operator cost, and the third component is the cost resulting from total travel demand excluding the transit demand satisfied by a specific network configuration. Note that C1, C2 and C3 are introduced to reflect the tradeoffs between the user costs, the operator costs and satisfied transit ridership, making the BTRNDP a multi-objective optimization problem. Generally, operator cost refers to the cost of operating the required buses. User costs usually consist of four components, including walking cost, waiting cost, transfer cost, and in-vehicle travel cost. The first constraint is the headway feasibility constraint, which reflects the necessary usage of policy headways in extreme situations. The second is the load factor constraint, which guarantees that the maximum flow on the critical link of any route r_m cannot exceed the bus capacity on that route. The third (fleet size) constraint represents the resource limits of the transit company and it guarantees that the optimal network pattern never uses more vehicles than currently available. The fourth constraint is the trip length constraint. This avoids routes that are too long because bus schedules on very long routes are too difficult to maintain. Meanwhile, to guarantee the efficiency of the network, the length of routes should not be too small. The fifth constraint is the maximum number of routes constraint, which reflects the fact that in solving the BTRNDP, transit planners often set a maximum number of routes, which is based on the fleet size. This has a great impact on the later driver scheduling work.

3 Tabu Search Algorithm

The TS algorithm has traditionally been used on combinatorial optimization problems and has been frequently applied to many integer programming, routing and scheduling, traveling salesman and related problems. The basic concept of TS is presented by Glover (1977), Glover (1986) who described it as a meta-heuristic superimposed on another heuristic. It explores the solution space by moving from a solution to the solution with the best objective function value in its neighborhood at each iteration, even in the case that this might cause the deterioration of the objective. (In this sense, "moves" are defined as the sequences that lead from one trial solution to another.) To avoid cycling, solutions that were recently examined are declared forbidden or "tabu" for a certain number of iterations and associated attributes with the tabu solutions are also stored. The tabu status of a solution might be overridden if it corresponds to a new best solution, which is called "aspiration." The tabu lists are historical in nature and form the Tabu Search memory. The role of the memory can change as the algorithm proceeds. Intensification strategies are based on modifying choice rules to encourage move combinations and solution features historically found good, and to initiate a return to attractive regions to search them more thoroughly. Diversification strategies are based on modifying choice rules to bring attributes into the solutions that are infrequently used, or to drive the search into new regions. Intensification and diversification are fundamental cornerstones of longer term memory in TS and reinforce each other. In many cases, various implementation models of the TS method can be achieved by changing the size, variability, and adaptability of the tabu memory to a particular problem domain. Basic versions of TS can be found in Glover (1989), Glover (1990), and variants ranging from simple to advanced can be found in Glover and Laguna (1997).

In all, TS is an intelligent search technique that hierarchically explores one or more local search procedures in order to search quickly for the global optimum. As one of the advanced heuristic methods, TS is generally regarded as a method that can provide a near-optimal or at least local optimal solution within a reasonable time for the BTRNDP. Details of our BTRNDP-specific TS algorithms are presented in Section 4.

4 Proposed Solution Methodology

The proposed solution framework consists of three main components: an *Initial Candidate Route Set Generation Procedure* (ICRSGP) that generates all feasible routes incorporating practical guidelines that are commonly used in the bus transit industry; a *Network Analysis Procedure* (NAP) that assigns the transit trips, determines the service frequencies on each route and computes many performance measures; and, a TS Procedure that combines these two parts, guides the candidate solution generation process and selects an optimum set of routes from the huge solution space. Fig. 1 gives the flow chart of the proposed solution framework. C++ is chosen as the implementation language in this research.

4.1 The Initial Candidate Route Set Generation Procedure (ICRSGP)

The ICRSGP configures all candidate routes for the current transportation network. It requires the user to define the minimum and maximum route lengths. The knowledge

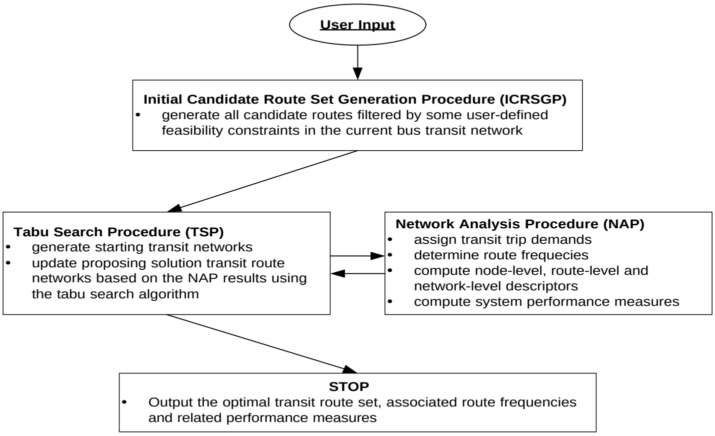

Fig. 1. Flow Chart of the Proposed Solution Methodology

of the transit planners has a significant impact on the initial route set skeletons, i.e., different user requirements result in different route solution space sets. ICRSGP relies mainly on algorithmic procedures including the shortest path and k-shortest path algorithms. Given the user-defined minimum and maximum length constraints, Dijkstra's shortest path algorithm (see Ahuja *et al.* (1993)) is used and Yen's k-shortest path algorithm (see Yen (1971)) is modified to generate all candidate feasible routes in the studied transportation network. Fig. 2 presents a skeleton for the ICRSGP.

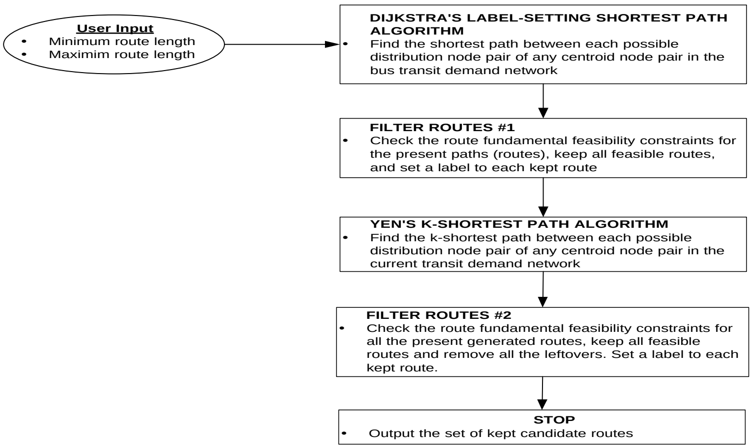

Fig. 2. Skeleton of the Initial Candidate Route Set Generation Procedure (ICRSGP)

4.2 The Network Analysis Procedure (NAP)

Fig. 3 shows the flow chart of the proposed network analysis procedure for the BTRNDP. Essentially, the NAP proposed in this paper is a bus transit network evaluation tool with the ability to assign transit trips between each centroid node pair onto each route in the proposed solution network and determine associated route frequencies. To accomplish these tasks for the BTRNDP, NAP employs an iterative procedure, which contains two major components, namely, a multiple transit trip assignment procedure and a frequency setting procedure, to seek to achieve internal consistency of the route frequencies.

Once a specific set of routes is proposed by the TS procedure in the overall candidate solution route set generated by the ICRSGP, the NAP is called to evaluate the alternative network structure and determine route frequencies. The whole NAP process can be described as follows. First, an initial set of route frequencies are specified because they are necessary before the beginning of the trip assignment process. Then, hybrid transit trip assignment models are utilized to assign the passenger trip demand matrix to a given set of routes associated with the proposed network configuration. The service frequency for each route is then computed and used as the input frequency for the next iteration in the transit trip assignment and frequency setting procedure. If these route frequencies are considered to be different from previous frequencies by a user-defined parameter, the process iterates until internal consistency of route frequencies is achieved. Once this convergence is achieved, route frequencies and several system performance measures (such as the fleet size and the unsatisfied transit demands) are thus obtained.

It should be noted that the trip assignment process considers each zone (centroid node) pair separately. Also, the transit trip assignment model presented in this paper adapts the lexicographic strategy (see Han and Wilson (1982)) and the previous transit trip assignment methods (see Shih *et al.* (1998)). However, several modifications have been made to accommodate more complex considerations for real world application. This model considers the number of transfers and/or the number of long walks to the bus station as the most important criterion. It first checks the existence of the 0-transfer-0-longwalk paths. If any path of this category is found, then the transit demand between this centroid node pair can be provided with direct route service and the demand is therefore distributed to these routes. If not, the existence of paths of the second category, i.e., 0-transfer-1-longwalk path and 1-transfer-0-longwalk paths are checked. If none of these paths is found, the proposed procedure will continue to search for paths of the third category, i.e., paths with 2-transfer-0-long-walk, 1-transfer-1-long-walk and/or 0-transfer-2-longwalks. Only if no paths that belong to these three categories exist, there would be no paths in the current transit route system that can provide service for this specific centroid node pair (i.e., these demands are unsatisfied). Note that at any level of the above three steps, if more than one path exists, a "travel time filter" is introduced for checking the travel time on the set of competing paths obtained at that level. If one or more alternative paths whose travel time is within a particular range pass the screening process, an analytical nonlinear model (i.e., the inverse proportional model) that reflects the relative utility on these

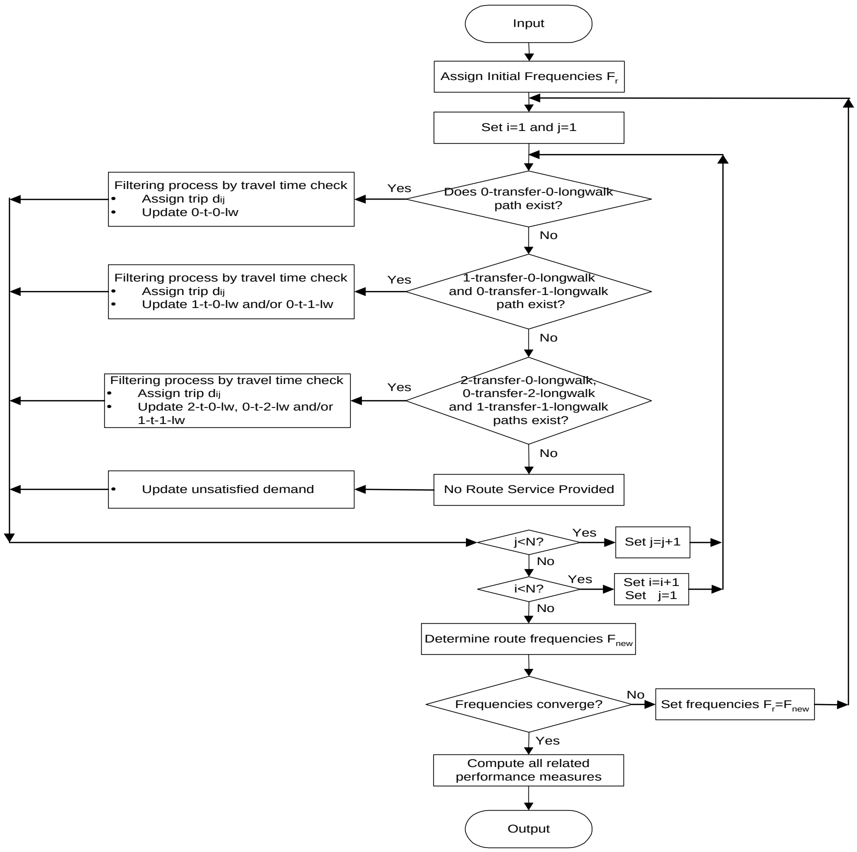

Fig. 3. Network Analysis Procedure (NAP) for the BTRNDP

competing paths is used to assign the transit trips between that centroid node pair to the network. In addition, policy headway and the demand headway are used together to determine the frequencies on each route in the frequency setting procedure. The whole process is repeated until all the travel demand pairs in the studied network are considered. Details of the transit trip assignment model can be seen in Fan and Machemehl (2004).

4.3 Tabu Search Procedure

Since the TS provides a robust search as well as a near optimal solution in a reasonable time, this approach is employed as one of the candidate solution techniques for BTRNDP. The following subsections present a systematic description for the TS algorithm-based implementation model for the BTRNDP.

Tabu Search Implementation Model: As with other heuristic algorithms, applying TS methods requires a significant amount of knowledge specific to the BTRNDP.

To make TS a potentially efficient algorithm for the BTRNDP, careful attention is required. Note that one of the significant contributions in this paper is using the TS algorithm to solve the BTRNDP. Since it is the first time for the TS methods to be applied for the BTRNDP, a detailed description of the BTRNDP-specific TS is presented.

Solution Representation: At any iteration t of the algorithm, let n represent the proposed solution route set size. A candidate bus transit route solution network can be represented by $X^t = (R_1^t, R_2^t, \ldots R_i^t, \ldots, R_n^t)$, where $R_i^t (i = 1, 2, \ldots, n)$ denotes the i-th bus route in the proposed solution set. Although the vector X^t is treated as ordered by the algorithm, it should be pointed out that X^t can also be treated as a set rather than a vector, and its ordering serves as a record keeping device for the algorithm rather than identifying a structural property of the solution itself. Let $f(X^t)$ represent the objective function as shown in the model formulation part for the proposed solution network defined by this n transit route network configuration $X^t = (R_1^t, R_2^t, \ldots, R_n^t)$.

Initial Solution: In this paper, all initial solutions for three different versions of the TS algorithms are randomly generated, with each solution being uniformly distributed in the solution space generated by the ICRSGP.

Neighborhood Structure: Undoubtedly, how to define the "neighborhood," i.e., the nearby solutions, might affect the quality of the transit route network solution. A different definition rule could result in a different solution of different quality. In this research, the neighborhood of a feasible solution route network set X^t is another feasible solution obtained by replacing one of the routes in the current proposed solution set, say the i-th route R_i^t to one of the routes that is next to R_i^t in the stored solution space. For route 1, the neighborhood can be defined as route 2 and route N, where N is the total number of routes in the stored solution space. For route N, the neighborhood can be defined as route 1 and route (N-1). The neighborhood of any route i $(1 < i < N - 1)$ that lies somewhere in the middle of the solution route space can be defined as the routes that are next to R_i^t. $Z(X_{ij}^t)$, the objective function value of a new solution X^{t+1} that is obtained from X^t by moving R_i^t to one of its neighbors R_j^t at generation t can be computed as follows: $Z(X_{ij}^t) = f(X^{t+1})$.

Moves and Tabu Status: As defined, a move consists of replacing a given route within X^t by one of its two neighboring routes that lie outside of X^t but within the stored solution space. It should be noted that both of these two neighboring routes are tried. At the beginning of this process, no move is tabu (i.e., forbidden). At any iteration with n number of routes in solution X^t, the algorithm executes the best non-tabu move out of $2 * n$ feasible moves to a feasible neighbor of the current solution. In addition, if a tabu move yields a worse solution which is, however, the best among all feasible neighbors of the current solution, it is also updated. Whenever a move is performed, the reverse move is declared tabu for m iterations, where m is either a user-defined parameter or a randomly generated one that follows a discrete uniform distribution in an interval $[m_{min}, m_{max}]$, where m_{min} and m_{max} are the user-defined minimum and maximum parameters of the algorithm. Comparisons of the model performance between these two strategies including the fixed and variable tabu tenure are performed in the numerical results part.

Diversification and Intensification: This part is developed to combine the diversification and intensification procedures to further explore the solution space for a possibly better solution. It starts from the best found solution route set and introduces a major perturbation by allowing q routes $(1 \leq q \leq n)$ to move w positions up from their current solution location (say q = 2 and w = 10) in the stored solution space. Put another way, X^t is moved to another feasible solution by replacing q routes within X^t by q other routes that each of them go up w position from their current solution location in the stored solution space. This is called "diversification." Note that this is a "forced" movement no matter whether the solution improves or not, so that the solution space can be somehow traversed more evenly. To respect the original characteristics of the TS, this procedure is never applied more than once during a given operation (called "intensification"). Note that tabu moves are also applied to this situation. If this move is toward one direction (say increasing direction) of the current route, then moves toward to the opposite direction (i.e., decreasing direction) are prevented for a certain number of iterations (say using the same m). Model performance comparisons of the TS algorithms between using and not using this procedure are also achieved and the better approach will be identified in the numerical results part.

Implementation Model Summary: In all, the proposed TS algorithms for the BTRNDP in this paper include two main procedures described as follows.

Neighborhood Search Procedure: At iteration t, let $X^t = (R_1^t, R_2^t, \ldots, R_n^t)$ be a feasible solution of value $f(X^t)$. Let $N(X^t)$ be the set of feasible neighbors of X^t, as defined before. The best neighbor of X^t is a solution $X_{i^*j^*}^t \in N(X^t)$ obtained by replacing one given route $R_{i^*}^t$ within X^t to its best neighbor $R_{j^*}^t$ that is one of its two neighboring routes outside X^t but within the stored solution space. Similarly define the best feasible non-tabu neighbor of X^t as $X_{\overline{ij}}^t \in N(X^t)$. ($X_{i^*j^*}^t$ and $X_{\overline{ij}}^t$ may coincide). Let X^* be the incumbent (the best known feasible solution) and let $Z(X^*)$ be its value.

If $Z(X_{i^*j^*}^t) < Z(X^*)$, set $X^* = X^{t+1} = X_{i^*j^*}^t$ and $Z(X^*) = Z(X^{t+1}) = Z(X_{i^*j^*}^t)$. Declare the move of a route from $R_{j^*}^t$ to $R_{i^*}^t$ tabu for m iterations, where m can be a fixed user-defined parameter or is uniformly distributed with $m \in [m_{min}, m_{max}]$. If $Z(X_{i^*j^*}^t) > Z(X^*)$ and all moves defining the solutions of $N(X^t)$ are tabu, set $\delta = 1$ and return. Otherwise, set $X^{t+1} = X_{\overline{ij}}^t$ and $Z(X^{t+1}) = Z(X_{\overline{ij}}^t)$. Declare the move of a route from R_j to R_i tabu for m iterations, where m has the same definition as used before.

Diversification and Intensification Procedure: This procedure is the same as that in Neighbor Search but defines $N(X^t)$ differently. It allows q routes $(1 \leq q \leq n)$ to move up to w more than the current solution location in the solution space (Note that in this paper, this procedure is called the "shakeup" procedure. Furthermore, for simplicity, q is set to n and w is set as a user-defined parameter). When a route is moved (i.e., replacing this route within X^t by another route that is w positions up/down from its current location in the stored solution space) in one direction (say the increasing direction), moving back in the opposite direction is declared tabu for m iterations, where m uses the same notation as before.

Tabu Search Algorithm for the BTRNDP:

Step 1 Randomly generate an initial feasible solution route network $X^t = (R_1^t, R_2^t, \ldots, R_n^t)$ with route size n in the proposed solution set.
Step 2 Set $\delta = 0, t = 1$ and $X^* = X^t$; While ($\delta = 0$ and $t \leq$ MAX_Iterations) Apply Neighborhood Search to the solution X^t; $t = t + 1$.
Step 3 Apply the "Diversification and Intensification" procedure to X^*. Apply Neighborhood Search to the solution X^* until $\delta = 1$ or $t >$ MAX_Iterations.
Step 4 Output the current best solution found.

As mentioned before, since TS provides a robust search as well as a near optimal solution within a reasonable time, this algorithm is employed as the solution technique for the BTRNDP. Before implementing the TS algorithms, a set of potential routes, consisting of the whole solution space, has been generated by the ICRSGP. The objective of the TS algorithm presented here is to select an optimal set of routes from the candidate route set solution space with the sum of the total user, operator and unsatisfied demand cost being minimized.

A flow chart that provides the typical TS algorithm-based solution framework for the BTRNDP can be seen in Fig. 4. Note that the "neighborhood" for any route i is defined as the route left or right of route i stored in the solution space, as described before. At the beginning of the TS implementation, the initial solution is randomly generated. In the second (and later) generation, the TS procedure is used to guide the generation of the new transit route solution set and after it is proposed at each generation, the search process is started. The network analysis procedure is then called to assign the transit trips between each centroid node pair and determine the service frequencies on each route and evaluate the objective function for each proposed solution route set. For each iteration, if a solution route set is detected to improve over the current best one, the current best solution is updated. The new proposed solution sets are generated and are evaluated in the same way. If convergence is achieved or the number of generations is satisfied, the iteration for a specific route set size ends. Then, the proposed solution route set size is incremented and the processes are repeated until the maximum route set size is reached. The best solution among all transit route solution sets is adopted as the best solution to the BTRNDP for the current studied network.

Moreover, in this paper, three versions of TS algorithms are used: 1) TS without shakeup procedure (i.e., without the diversification procedure as defined before) and with fixed tabu tenures; 2) TS with shakeup procedure and fixed tabu tenure (i.e., the number of restrictions set for the tabu moves are fixed); and 3) TS with shakeup procedure and variable tabu tenure (i.e., the number of restrictions set for the tabu moves are randomly generated). The differences underlying each TS algorithm are self-explanatory by the names. All three variations of TS methods are implemented, sensitivity analysis for each version are presented, and algorithm comparisons are performed.

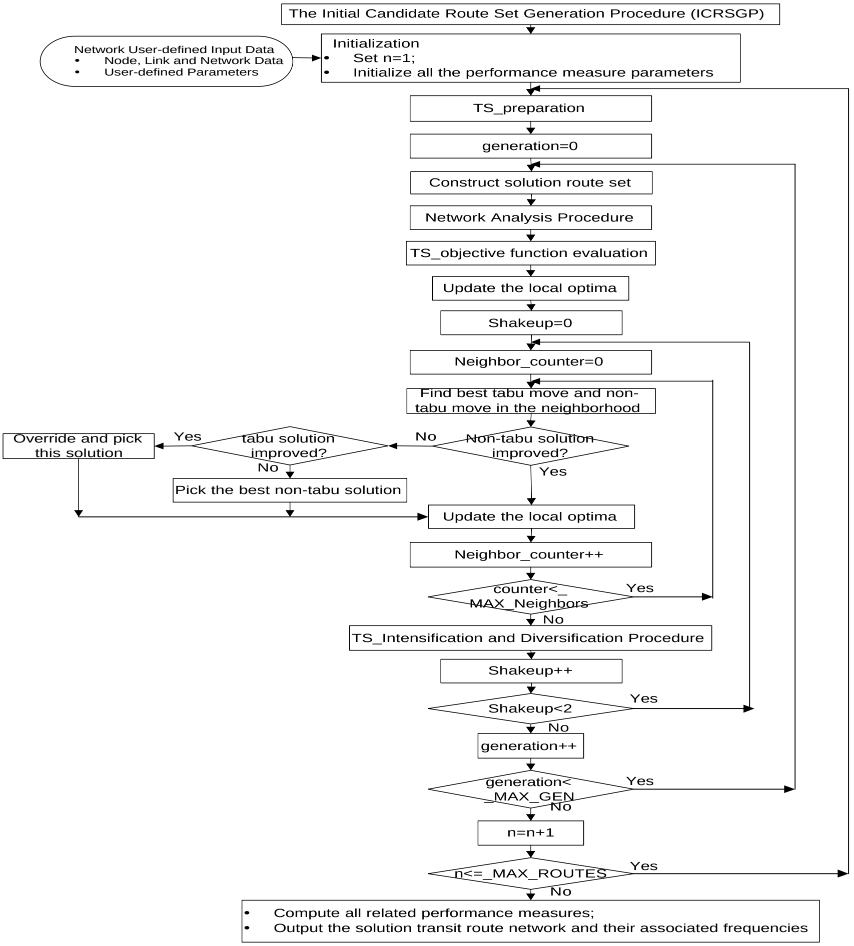

Fig. 4. A Tabu Search Model Based Solution Framework for the BTRNDP

5 Experimental Network and Numerical Results

5.1 Example Network Configuration

The TS algorithm-based solution methodology is implemented using a small example network as shown in Fig. 5. This example network contains seven travel demand zones and 15 road intersections. As noted before, the ICRSGP discussed in this paper first considers the BTRNDP under the “centroid” level. The network is processed as follows: 1) the zonal demands are distributed the same way as the highway network demand; and 2) if the same road link contains two or more demand distribution nodes from different zones, these distribution nodes are aggregated. After this preliminary process, 20 centroid distribution nodes, 35 nodes, and 82 arcs are obtained in this example network. The minimum and maximum route lengths are defined. In the example first phase, the ICRSGP generates 286 feasible routes whose distances satisfy two route length constraints as mentioned before.

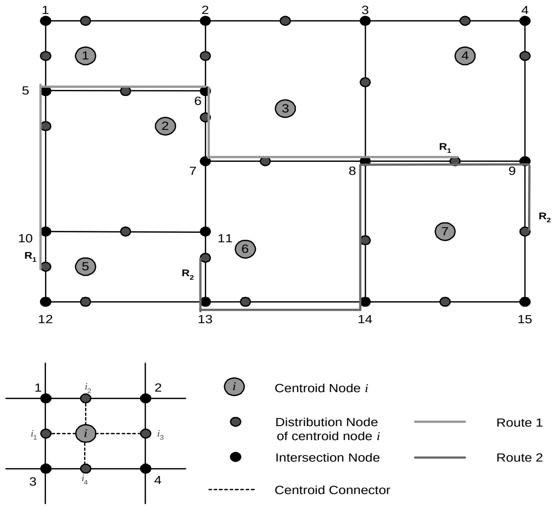

Fig. 5. A Small Network With Graphical Representations for Nodes, Links and Routes

5.2 Numerical Results and Sensitivity Analysis

It is noted that the performance of the proposed TS algorithms might greatly depend upon the chosen parameters such as the number of generations, the number of search neighbors, the number of tabu tenures and the shakeup number. Furthermore, note that since these parameters are basically continuous, one has to get the “nominally” optimal parameter through sequential testing. In addition, since the objective function is a multi-objective decision making problem, a commonly used weight set (0.4, 0.4 and 0.2) is assigned to each of the three objective function components (user cost, operator cost and unsatisfied demand cost), respectively, for demonstrating the sensitivity analysis here. Fig. 6 presents the sensitivity analysis of these parameters using the tabu algorithm without shakeup and with fixed tenures as an example. The effect of generations, tabu tenures and search neighbors are examined by varying these values within a specific range, and the results are given from Fig. 6.1 to 6.3, respectively. Details are described as follows.

Effect of Generations: Basically, “Generation” is a user-defined parameter which means how many iterations the transit planners want the developed solution algorithm run. It therefore can be varied from 1 to ∞. However, for efficiency, the effect of the number of generations is examined by varying this value from 5 to 100 and the result is given in Fig. 6.1. It can be seen from the figure that as the number of generations increases, the objective function value tends to decrease. It is also noted that the larger the chosen number of generations, the more the computation time. When the number of generations reaches 30, the optimal objective function is achieved, suggesting that 30 should be chosen as the optimal generations for the small network. Therefore, a generation of 30 was recommended.

Effect of Tabu Tenures: The effect of tabu tenures (i.e., the number of restrictions) is investigated by choosing this number ranging from 5 to 40 and the result is provided in Fig. 6.2. As can be seen, the least objective function value occurred with ten restrictions. Therefore, ten is chosen as the best number of tabu move tenures.

Effect of Search Neighbors: The effect of search neighbors is also studied by varying this value from 10 to 100. The result shown in Fig. 6.3 indicates that 20 might be the best value and as a result, it is recommended.

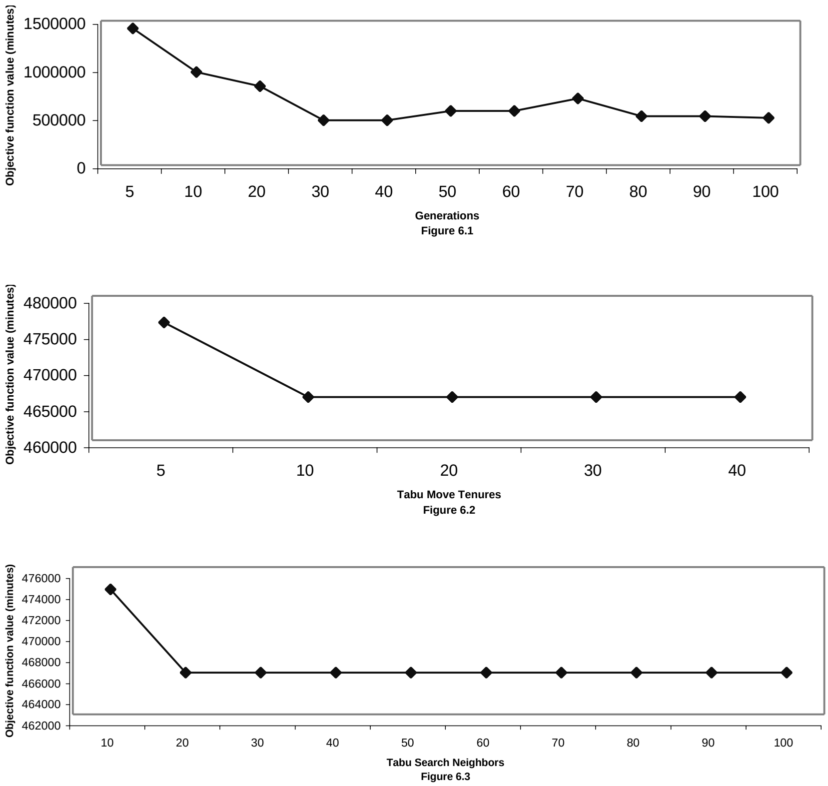

Fig. 6. Sensitivity Analysis for the Tabu Algorithm Without Shakeup and With Fixed Tenures

The above subsections presented the sensitivity analysis for tabu algorithm without shakeup and with fixed tenures using the example network. For sensitivity analysis regarding the other two developed TS methods including the tabu with shakeup/fixed tenures and that with shakeup/variable tenures, similar procedures can be followed. In addition, the genetic algorithm is used as a benchmark in this paper to examine the solution quality obtained from these three TS algorithms. The sensitivity analysis are also performed for the genetic algorithm using the same procedure (details about the genetic algorithm implementation model can be seen from Fan and Machemehl (2004)). Table 1 provides a summary of these sensitivity analysis for each algorithm for the BTRNDP. The best parameter set for each algorithm thus can be seen and chosen.

Table 1. Summary of Algorithm Sensitivity Analysis for the BTRNDP

Genetic Algorithm		Population Size	30
		Generations	20
		Crossover Probability	0.8
		Mutation Probability	0.1
Tabu Search	Tabu w/o Shakeup and with Fixed Tenures	Generations	30
		Tenures	10
		Search Neighbors	20
	Tabu w/t Shakeup and Fixed Tenures	Generations	80
		Tenures	10
		Search Neighbors	10
		Shakeup Number	50
	Tabu w/t Shakeup and Variable Tenures	Generations	20
		Search Neighbors	40
		Shakeup Number	50

5.3 Multi-Objective Decision Making and Algorithm Comparisons

As mentioned, the model performance based on each proposed algorithm might greatly depend upon the chosen value of parameters inherent in that algorithm. In previous sections, a set of user-defined parameters associated with each algorithm is found by first assigning a commonly used weight set to each of the three objective function components and then running the developed programming codes based on that algorithm several times. The sensitivity analysis are then performed and the best parameter set is found by choosing those resulting in the least objective value from that algorithm. In this section, these chosen parameters for each algorithm are used and applied to the BTRNDP at different chosen weight levels. The objective is to see how the quality of these algorithms varies across different weight levels and one might therefore know which algorithm can be used to best solve the BTRNDP. The following sections compare the three employed TS algorithms to examine which variation is most suitable for the BTRNDP. Furthermore, the model performance is also compared to the genetic algorithm as a benchmark to examine the solution quality using TS algorithms from a multi-objective decision making perspective.

Fig. 7 presents numerical results for these comparisons using the example network. For each graph, the weight of total unsatisfied demand cost is set at a specific level between 0.1 and 0.8. The x-axis denotes the weight of total user cost and the y-axis is the objective function value measured in minutes. Note that each point shown for each algorithm in each graph is a decision making problem with a particular weight set for the three components contained in the objective function, where the weight of total operator cost can be obtained at each point by subtracting 1.0 from the weight sum of total unsatisfied demand cost and user cost. One can see from Fig. 7 that TS with shakeup and fixed tenures (i.e., fixed iterations) clearly seems to outperform other TS algorithms using the example network at any weight set level. Therefore, this tabu algorithm is chosen as the best TS algorithm for the BTRNDP.

It can also be seen from Fig. 7 that for each algorithm from any graph, as the weight of total user cost increases, the objective function value obtained by using that algorithm tends to increase. This is expected because the user cost is usually greater than the operator cost and the increase in total user cost due to a 0.1 unit increase in the weight of total user cost outweighs the decrease in total operator cost due to a 0.1 unit decrease in the weight of total operator cost. As a result, the total objective function value increases. One interesting phenomenon is that the genetic algorithm seems to be more variable than any TS algorithm (except the TS with shakeup and with variable tenures, which is also variable due to its inherent variable nature underlying the tabu tenures) in terms of the optimal objective function value (from Fig. 7.1 to 7.5.) This might suggest that, compared to TS algorithms, the Genetic Algorithm (GA) may largely depend on the chosen parameters at any particular level. If the chosen parameters inherent in the GA are fixed, the solution quality for the BTRNDP might be unstable. Therefore, to achieve the best solution network at each weight set level, one might need to run the program and get the optimal parameter set at that level although the computational burden would become larger. Furthermore, for each graph (i.e., for each weight level for the total unsatisfied demand cost), the TS with shakeup and fixed tenures seems to consistently outperform the GA in terms of the quality of solution (i.e., it always results in the least objective function value). This might allow the conclusion that compared to the GA, this TS method performs better for solving the BTRNDP. Furthermore, it can be seen that the local optimal solution obtained from this TS method can provide solution of very high quality because it is very near to the global optimum. The GA, however, seems to be the undesirable model. This might be possible because although the GA might achieve some better solutions by learning from the previous solutions through a genetic approach, it might take much more time inside the algorithm itself to look for this achievement, while it does not take much more effort looking for possibly better solutions from other "neighborhood" solutions in the candidate solution space (compared to the TS algorithms). Conversely, the TS with shakeup and fixed tenures not only can look for a good solution with a specific origin-destination node pair through "random search" in its early stage, but also can fully explore possibly better neighborhood solutions. Note that the tradeoffs between route coverage and the route directness might be well balanced between chosen shortest paths or k-th shortest paths between specific origin-destination node pairs. It is expected that this inherent characteristics of the TS algorithm might make it particularly suited for the BTRNDP and therefore outperform the GA.

5.4 Characteristics of the BTRNDP

The characteristics of the BTRNDP are very extensive due to its multi-decision making nature and the variety of parameters and procedures involved. These characteristics might depend upon the network size, the chosen parameters in the solution process, the chosen algorithm and the chosen weight level for each component of the objective function. In this sense, it is very hard to generalize all characteristics of the BTRNDP. However, it is expected that in most cases, the BTRNDP characteristics should be similar and the current comprehensive numerical results also show these similarities. Since the numerical results based upon weights of 0.4, 0.4 and 0.2 for the user cost, operator cost and unsatisfied demand cost, respectively, using the tabu algorithm without shakeup and with fixed tenures seem to be very representative, these are chosen here for presenting related BTRNDP characteristics.

The effect of the number of proposed routes in the transit network solution is investigated by varying it from 1 to 10 and the values of each performance measure of the optimal network at each route set size level including the user cost, the operator cost, the fleet size required, the unsatisfied demand cost, the percentage of the satisfied transit demand and the total objective function value are shown in Fig. 8.1 through 8.6, respectively. Generally speaking, as the number of routes provided in the network increases, more passengers will be served by transit and therefore, the satisfied transit demand increases. Furthermore, since the fixed transit demand is assumed, the percentage of satisfied transit demand also tends to increase as shown in Fig. 8.5. Also as a result, the unsatisfied demand cost decreases. However, the operator cost tends to increase because the fleet size required for the network generally increases. In addition, the user cost generally increases because more transit users travel and the total objective function value also increases. The reason might be that although service might be better in some sense (such as more passengers get direct route service) as more routes are provided, the headway might be longer on some routes. Therefore, the transit user cost as a whole might actually increase. In conclusion, the numerical results in Fig. 8 indicate that as a whole, as the route set size increases, the solution improved initially because more demand was satisfied and unsatisfied demand costs decrease. However, the least objective function value is achieved with two routes for this scenario and increases in the fleet size (i.e., operator cost) produces underutilization of routes and does not result in an improved objective function value. (Note that the optimal transit route network is shown in Fig. 5.)

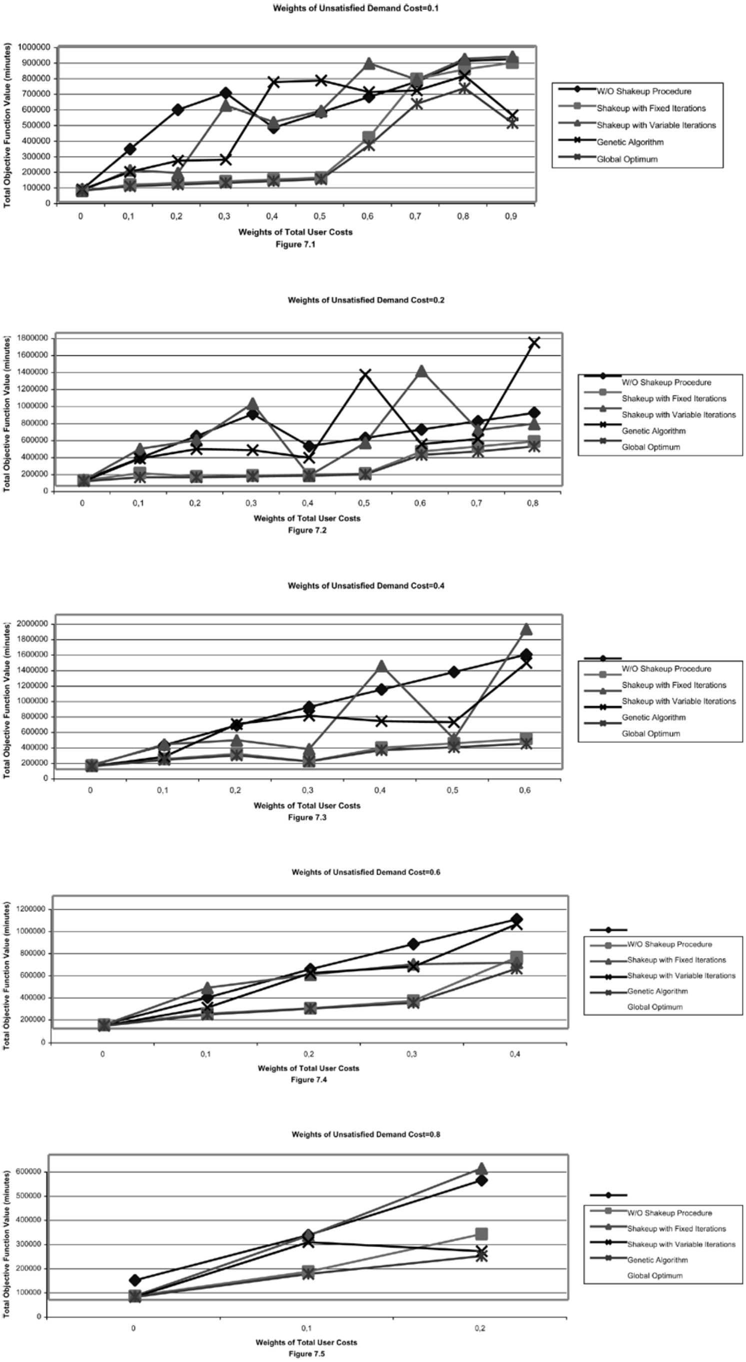

Fig. 7. TS and GA Comparisons for the BTRNDP

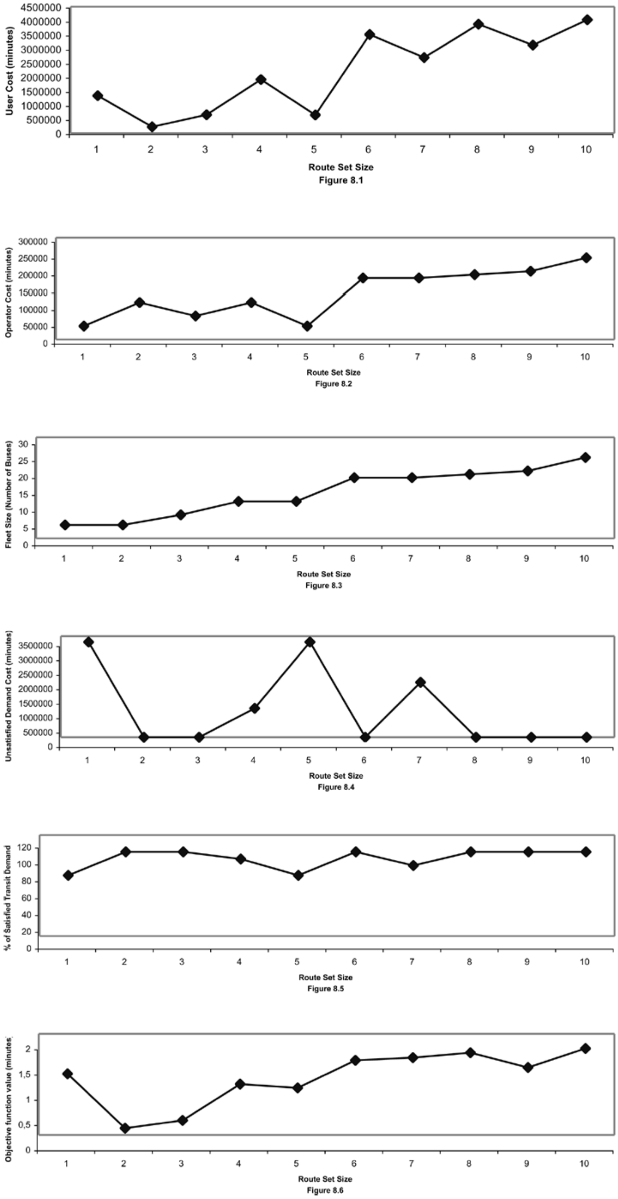

Fig. 8. Effect of Route Set Size on Objective Function and its Components for the BTRNDP

6 Conclusions

This paper uses TS algorithms to solve the optimal bus transit route network design problem at the distribution node level. A multi-objective nonlinear mixed integer model is formulated for the BTRNDP. The proposed solution framework consists of three main components: an Initial Candidate Route Set Generation Procedure that generates all feasible routes incorporating practical bus transit industry guidelines; and a Network Analysis Procedure that assigns transit trips, determines service frequencies and computes performance measures; and, a TS procedure that guides the candidate solution generation process. Three different variations of TS algorithms are employed and compared as the solution method for finding a hopefully optimal set of routes from the huge solution space. A C++ program is developed to implement the TS algorithms for the BTRNDP. A small example network is successfully tested as a pilot study. The model comparisons are performed and numerical results are presented. The TS with shakeup and fixed tenures is identified as the best TS method to solve the BTRNDP. A genetic algorithm is also used as a benchmark to measure the quality of the TS methods and numerical results clearly indicate that the preferred TS method outperforms the genetic algorithm using the example network. Furthermore, the local optimal solution obtained from this TS method can provide solutions of very high quality because it is very near to the global optimum. In addition, related characteristics and tradeoffs underlying the BTRNDP are also discussed.

BTRNDP is a really complex problem. One simple neighborhood rule can be the swapping of nodes. However, the link connectivity problem can make many routes resulting from swapping infeasible. Although one can always find routes to connect any two nodes to make it feasible, the efficiency can be a big problem. One option for future investigation is to examine a more flexible neighborhood definition that allows replacement by non-adjacent routes and the tabu status would then refer to forbidding the re-instatement of specific routes for a given period. Another possibility that may be worth mentioning is the investigation of a different type of short term memory that recent investigations have shown effectiveness (Glover and Laguna (1997)). Also, further application of this model to a very large network is under the way.

Acknowledgements: The authors want to express their deepest gratitude to two anonymous reviewers for their incisive and seasoned suggestions. The authors also appreciate the U.S. Department of Transportation, University Transportation Center through SWUTC to the Center for Transportation Research, The University of Texas at Austin for sponsoring this research by Projects 167525 and 167824.

References

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, Englewood Cliffs.

Baaj, M. H. and Mahmassani, H. S. (1992). Artificial intelligence-based system representation and search procedures for transit route network design. *Transportation Research Record 1358, Transportation Research Board*, pages 67–70.

Ceder, R. B. and Wilson, N. H. (1986). Bus network design. *Transportation Research*, **20B**(4), 331–344.

Chien, S., Yang, Z., and Hou, E. (2001). A genetic algorithm approach for transit route planning and design. *Journal of Transportation Engineering, ASCE*, **127**(3), 200–207.

Fan, W. and Machemehl, R. B. (2004). *A Genetic Algorithm Approach for the Transit Route Network Design Problem, CSCE 2004, 5th Transportation Specialty Conference*. Saskatoon.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences*, **8**(1), 156–166.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, **5**, 533–549.

Glover, F. (1989). Tabu search, part I. *ORSA Journal on Computing*, **1**, 190–206.

Glover, F. (1990). Tabu search, part II. *ORSA Journal on Computing*, **2**, 4–32.

Glover, F. and Laguna, M. (1997). *Tabu Search*. Kluwer Academic Publishers.

Han, A. F. and Wilson, N. (1982). The allocation of buses in heavily utilized networks with overlapping routes. *Transportation Research*, **16B**, 221–232.

NCHRP Synthesis of Highway Practice 69 (1980). Bus route and schedule planning guidelines. Technical report, Transportation Research Board, National Research Council, Washington, D.C.

Pattnaik, S. B., Mohan, S., and Tom, V. M. (1998). Urban bus transit network design using genetic algorithm. *Journal of Transportation Engineering*, **124**(4), 368–375.

Shih, M., Mahmassani, H. S., and Baaj, M. (1998). Trip assignment model for timed-transfer transit systems. *Transportation Research Record 1571*, pages 24–30.

Yen, J. Y. (1971). Finding the k shortest loopless paths in a network. *Management Science*, **17**(11), 712–716.