Application of A Route Expansion Algorithm for Transit Routes Design in Grid Networks

Iran Khanzad[1], Amirali Zarrinmehr [2], Seyedehsan Seyedabrishami [3], Mahmoud Saffarzadeh [4]

Received: 01. 02. 2016 Accepted: 17.05.2016

Abstract

Establishing a network of transit routes with satisfactory demand coverage is one of the main goals of transit agencies in moving towards a sustainable urban development. A primary concern in obtaining such a network is reducing operational costs. This paper deals with the problem of minimizing construction costs in a grid transportation network while satisfying a certain level of demand coverage. An algorithm is proposed following the general idea of "constructive algorithms" in related literature. The proposed algorithm, in an iterative approach, selects an origin-destination with maximum demand, generates a basic shortest-path route, and attempts to improve it through a route expansion process. The paper reports the scenarios and further details of the algorithm considered for expanding a transit route in a grid network. A random 6×10 grid network is applied to report the results. The results support that application of the proposed algorithm notably reduces the operational costs for various amounts of demand coverage.

Keywords: Transit routes, grid transportation network, demand coverage, operational costs

---

Corresponding author E-mail: saffar_m@modares.ac.ir

1. MSc Grad., Department of Civil and Environmental Engineering, Tarbiat Modares University, Tehran, Iran
2. Ph.D Candidate, Department of Civil and Environmental Engineering, Tarbiat Modares University, Tehran, Iran
3. Assistant Professor, Department of Civil and Environmental Engineering, Tarbiat Modares University, Tehran, Iran
4. Professor, Department of Civil and Environmental Engineering, Tarbiat Modares University, Tehran, Iran

1. Introduction and Background

In many modern cities, urban transportation is confronted with serious challenges such as traffic congestion, air pollution, and parking shortages. These problems have been recognized as the impacts of excessive use of private modes of transportation. To resolve such issues and move towards a sustainable transportation, it is essential to design and implement an affordable public transportation system, namely transit system, offering high-quality services to the network users [Guihaire and Hao, 2008; Kepaptsoglou and Karlaftis, 2009].

Designing a complete transit system poses a very large problem which is often divided in smaller tractable sub-problems to solve [Cancela, Mauttone, and Urquhart, 2015]. Consequently, decision making in transit planning has been categorized in different levels from long-term decisions on urban infrastructures to short-term ones such as vehicle/crew scheduling [Guihaire and Hao, 2008]. In one of the most fundamental levels of decision making, Transit Routes Design (TRD) is concerned with determining an optimal set of transit routes in a public transportation system, e.g. bus or rail networks. Since almost all decisions in transit planning are conditioned to the configuration of routes, TRD plays a significant role in the process of generating an efficient transit system [Ceder and Wilson, 1986].

TRD is a hard problem to solve [Schöbel, 2012]. At least five sources of complexity for this problem have been mentioned in the study of Baaj and Mahmassani [1991] as follows: (1) Difficulty in defining decision variables and objective function; (2) Non-linearities and non-convexities exhibited by formulations; (3) Combinatorial explosion of the problem; (4) Multi-objective nature of the problem; and (5) Characterizing a “good” spatial layout for the routes. Due to such difficulties arising in TRD problem, most of the studies have adopted heuristic or meta-heuristic solution approaches to tackle the problem. A categorization of these solution approaches can be found in review papers such as Kepaptsoglou and Karlaftis [2009] or Farahani et al [2013].

Various objective functions have been applied so far to address the problem of TRD. The difference in objective functions mainly stems from the different levels of decisions in transit planning. Most review papers, such as Farahani et al [2013] or Ibarra-Rojas et al [2015], generally categorize these decisions in three levels: (1) Strategic decisions, e.g. designing or expanding transit routes; (2) Tactical decisions, e.g. allocating exclusive bus lanes or service frequency determination; and (3) Operational decisions, e.g. scheduling problems. Since TRD falls into the category of strategic planning problems, it is natural to ignore many unnecessary details and take into account only the main modeling aspects of the problem.

Two important features of TRD problem at strategic level are transit operational costs and users’ benefits. To model these features of the problem, many authors have used the total duration (i.e. length/travel-time) of transit routes and direct coverage, respectively. The total duration of transit routes is a representative of the operational costs in a sense that construction costs and transit fleet size are directly proportional to it [Mauttone and Urquhart, 2009]. Also, direct coverage stands for the percentage of the travel demand which can be satisfied with no transfers between transit routes and is a representative of users’ benefit. This is supported by the observation that as soon as the number of transfers increase the transit users switch to private modes of transportation [Stern, 1996; Guihaire and Hao, 2008].

Generating transit routes in TRD often consists of building shortest-paths and then improving them by adding further network links [Mandl, 1980; Kepaptsoglou and Karlaftis, 2009]. Among various studies over TRD problem in which a certain level of demand coverage is ensured and the total duration of routes is minimized, there are only a few algorithms that build the routes set from scratch [Mauttone and Urquhart, 2009]. These algorithms, also known as “constructive” algorithms, have been the topic of research in Baaj and Mahmassani [1995] and Mauttone and Urquhart [2009].

The general idea of constructive algorithms in TRD is to generate basic shortest-path routes between node-pairs and improve the coverage of routes by marginally modifying their layout. In the study of Baaj and Mahmassani [1995] a route generation process is proposed in which node-pairs with maximum demand values are iteratively selected and a transit route is generated between the selected nodes. To cover more transit trips, the authors take the advantage of the network nodes that are close to the generated routes. To this end, they propose a "node insertion" procedure in which transit nodes at distance of 1-link from generated routes are examined to improve the routes layout. Mauttone and Urquhart [2009] also proposed a new constructive algorithm based on the inter-zonal nature of demand in the network. Instead of inserting a single node in a transit route, they proposed to insert a pairs of nodes (in form of a path) into the set of routes. This approach was reported to significantly reduce the total duration of transit routes in the problem.

The above-mentioned constructive algorithms have been suggested as general-purpose approaches in the literature. Designing constructive algorithms and investigating their performances over special networks, to the best of our knowledge, have not been reported. Different attributes are recognized in the literature for transportation networks with special structures, such as grid or radial networks [Badia, Estrada and Robusté, 2014; Snellen, Borgers, and Timmermans, 2002]. As Figure 1 illustrates, the outcome of node insertion in a grid network may be different from a general transportation network. It can be observed from this figure that, for example, after insertion of the node n in the transit route, the route directness (between nodes i and j) and duration in the grid network may turn to be more degraded than the general network.

In this study, we present a detailed description of a constructive algorithm and corresponding results for grid transportation networks. The general scheme of the algorithm is based on the idea of node insertion in constructive algorithm [Baaj and Mahmassani, 1995] whereas the details are designed for a grid network. The paper investigates whether or not, and to what extent, the proposed constructive algorithm can improve the layout of shortest-paths in a grid network. In the following sections, the problem is described in section 2 and the solution algorithm is presented in section 3. The functionality of the algorithm and numerical results are discussed in sections 4 and 5 respectively. The paper is finally conclude in section 5.

2. Description of the Problem

The problem of Transit Routes Design (TRD) deals with determining the set of transit routes in an urban transportation network. The set of routes must meet the requirements of both network users and operators [Mauttone and Urquhart, 2009]. To model the benefits of network users, many authors have used the demand coverage offered by transit routes (see for example Zhao and Zeng [2006] or Farahani et al [2013]). The demand coverage is defined in this paper as the percentage of commuters that can make their trips by transit routes with no transfers. Also, the total length of transit routes has been widely applied in the literature as an index for the operating costs. The two mentioned measures, namely demand coverage and total length of transit routes, are applied in this paper to address the requirements of network users and operators, respectively.

There are two main approaches in TRD literature dealing with demand coverage and operational costs. Some studies consider the operational costs within a constraint of the problem and attempt to maximize the coverage [Kermanshahi, Shafahi, and Bagherian, 2015], while others take the coverage as a constraint and minimize the operational costs [Mauttone and Urquhart, 2009]. The latter approach is adopted in this paper to solve the TRD problem. We assume that a grid-type network of highways and the corresponding matrix of travel demand are available. The demand matrix and also link travel-times are assumed to be fixed in this study. It is intended to minimize the operational costs while a minimum demand coverage must be satisfied.

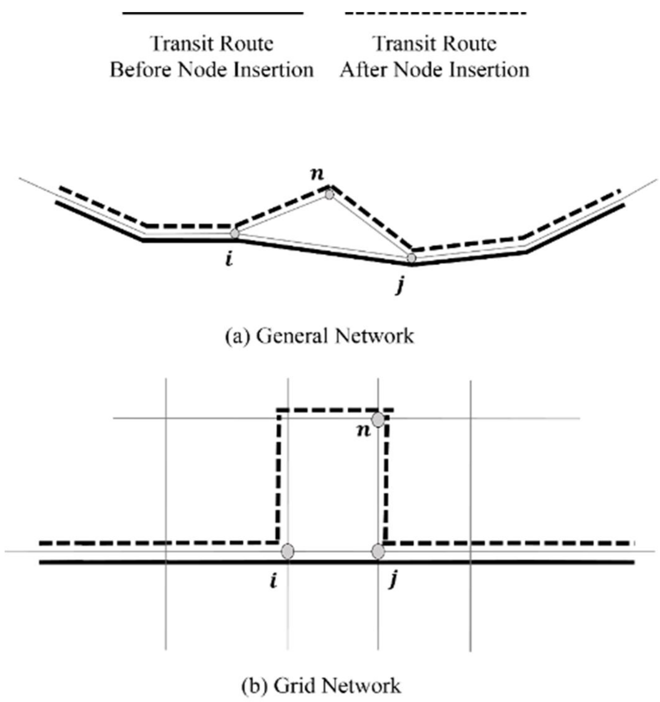


Figure 1. **Node insertion in a general network and a grid network**

To describe the problem in terms of a mathematical model, let us define:

GS: the global set of all feasible transit routes between origin-destinations in the entire network;

R: the set of selected transit routes from GS;

r: a transit route from *R*;

t(*r*): the travel time (i.e. length) associated with the transit route *r* (minutes);

Cov(*R*): demand coverage for the set of routes *R* (%);

Cov_{min}: the minimum demand coverage to be satisfied by selected routes (%).

Based on these definitions, a global formulation for TRD problem may be written as follows:

$$\underset{R}{\text{Min}} \sum_{r \in R} t(r) \quad (1)$$

$$R \subseteq GS \quad (2)$$

$$\text{Cov}(R) \geq Cov_{min} \quad (3)$$

In the above formulation, the objective function (1) aims at minimizing the total travel time of transit routes, namely R. Constraint (2) states that the set of selected routes is a subset of the global set of all feasible routes in the network and constraint (3) ensures that demand coverage will not be lower than a predefined value i.e. the minimum coverage.

In the formulation (1)-(3), the exponential number of alternative sets of transit routes, R, renders the problem computationally intractable. This fact may be better observed through a simple analysis in Figure 2.

Figure 2 indicates that the number of alternative subsets, R, to be selected in a transportation network with p pairs of origin-destinations and q feasible routes on average for each pair is in the order of 2^{pq}. For example, considering a transportation network with 10 nodes, which may be looked upon as a toy example in comparison with real-world instances, the number of origin-destinations is p=10×9=90.

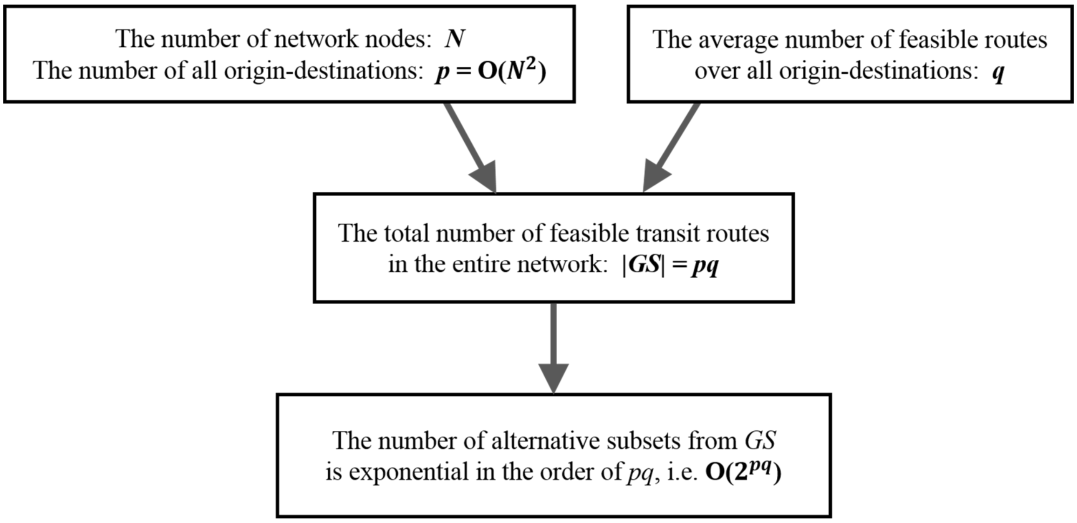


Figure 2. A simple analysis for the size of the solution space in TRD problem

Given that only q=2 feasible candidate routes exist on average for each origin-destination, there will be the number of $|GS|$=90×2=180 routes to be selected at the entire network. Now, the total number of subsets that may be selected from these feasible routes is clearly of the order 2^{180} which is an extremely-large number. Such an order is clearly a hindrance to the exact solution of the problem (1)-(3). Application of non-exact (e.g. heuristic) solutions algorithms, as a result, is inevitable to tackle this problem.

3. The Proposed Algorithm

This section presents a heuristic algorithm to solve the TRD problem. The general idea of the algorithm is based on the route generation algorithm proposed by Baaj and Mahmassani [1995]. However, the focus is put on designing a detailed algorithm for a grid transportation network as one of the regular structures used in modern cities [Badia, Estrada and Robusté, 2014]. Further, since there are many details arising at the stage of implementing the algorithm, some most important details of the algorithm will be reported.

The flowchart of Figure 3 sketches the main components of the proposed algorithm. The notations required beforehand are as follows: N: the set of all nodes in the network;

P: the set of all origin-destinations in the network, $P = N \times N$;

R: the set of all transit routes generated as the algorithm proceeds;

r: a candidate transit route;

t^*_{ij}: the shortest-path travel-time between nodes i and j (minutes), $i,j \in N$;

$t_{ij}(r)$: the travel-time between i and j within route r (minutes), $i,j \in N$, $r \in R$;

ω: an upper bound for the ratio of $t_{ij}(r)/\,t^*_{ij}$, $i,j \in N$, $r \in R$;

dis(n,r): the minimum number of links intervening between node n and route r, $n \in N$, $r \in R$;

Adj(r): the set of nodes, such as n, for which dis(n,r)=1, $r \in R$;

Cov(n,r): the total amount of travel demand between node n and all nodes in route r (passengers per hour), $n \in N$, $r \in R$;

S(r,n): the set of all candidate expansions of route r covering node n, $n \in N$, $r \in R$;

$S^a(r,n)$: the set of all acceptable candidate expansions of route r covering node n, $S^a(r,n) \subseteq S(r,n)$, $n \in N$, $r \in R$;

Common(r,R): the number of common links between a new transit route r and previously generated routes R;

λ: a maximum value for Common(n,r);

The algorithm, as shown in Figure 3, builds the set of routes R from scratch in an iterative approach. Following the study of Baaj and Mahmassani [1995], in each iteration, the algorithm generates a shortest-path route to cover a new origin-destination with maximum demand. Then, in a route expansion loop which is designed for grid networks, the algorithm attempts to further cover the most promising nodes that are close to the generated route. The process of route generation finishes after the minimum coverage constraint is satisfied during the iterations.

In what follows, the main components of the algorithm are discussed in more details.

Initial Settings: The algorithm starts with an empty set of transit routes and initially sets the corresponding coverage to zero.

Minimum Coverage Condition: This condition checks whether the set of routes built so far satisfies the minimum demand coverage or not. If the minimum demand was covered, the algorithm is terminated by printing the routes' information in the output. Otherwise, an iteration is started to generate a new route.

Building a Basic Transit Route: A basic transit route is built by picking up the origin-destination with maximum demand and generating the associated shortest-path. The set of nodes that are adjacent to the route r, namely Adj(r), is also generated. These nodes serve as candidate nodes to be covered along with the route r.

Route Expansion Loop: The loop of route expansion is a main part of the algorithm in which the transit route r is iteratively expanded and updated until the list of candidate nodes to be covered becomes empty. In this loop, first, a candidate node from Adj(r) is selected. Then, alternative route expansions are built to cover the selected node, and finally, the route is updated.

Finding a Candidate Node: In a greedy approach, the algorithm selects the node n with maximum value of Cov(n,r) in each iteration of route expansion loop. It is obvious that covering such node is more likely to increase demand coverage.

To define alternative expansions of route r covering the node n, i.e. set $S(r,n)$, let denote the node adjacent to n in route r by n_i. Moreover, let n_{i-1} and n_{i-2} be the two nodes of route r lying just before node n_i and also n_{i+1} and n_{i+2} be the two nodes just after that, as depicted in Figure 4 (a). Based on these definitions, the algorithm generates alternative sub-routes between nodes $\{n_{i-2}, n_{i-1}, n_i\}$ and $\{n_i, n_{i+1}, n_{i+2}\}$ to cover the node n.

To make sure that the sub-routes generated in the route expansion loop will offer new transit configurations, first, the existing transit sub-routes are removed from the network. Then, shortest-paths including node n are used to generate alternative sub-routes between pairs of (n_{i-1},n_i), (n_{i-2},n_i), (n_i,n_{i+1}), (n_i,n_{i+2}), (n_{i-1},n_{i+1}), (n_{i-2},n_{i+2}), (n_{i-2},n_{i+1}), (n_{i-1},n_{i+2}). Figure 4 shows a simple topology of a transit route in a grid network and alternative route expansions therein.

Two constraints are also considered for alternative route expansions, which are defined as route acceptability conditions in the flowchart of Figure 3. The first condition ensures that an alternative route expansion, temporarily named as *r_temp*, does not violate the minimum or maximum lengths as predefined standards of transit routes. The second condition states that the travel-time between each two nodes of *r_temp* must be not more than ω-times the shortest travel-time between those nodes. The latter condition ensures a certain level of route directness between any pairs of transit nodes in a transit route.

Update the Route and Adjacent Nodes: Among alternative route expansion (including the first route itself), it is desirable to select the candidate route with maximum demand coverage and minimum length. Since these two goals may be conflicting with one another, the measure Cov(*r_temp*)/t(*r_temp*) is used in the algorithm to select the final route expansion.

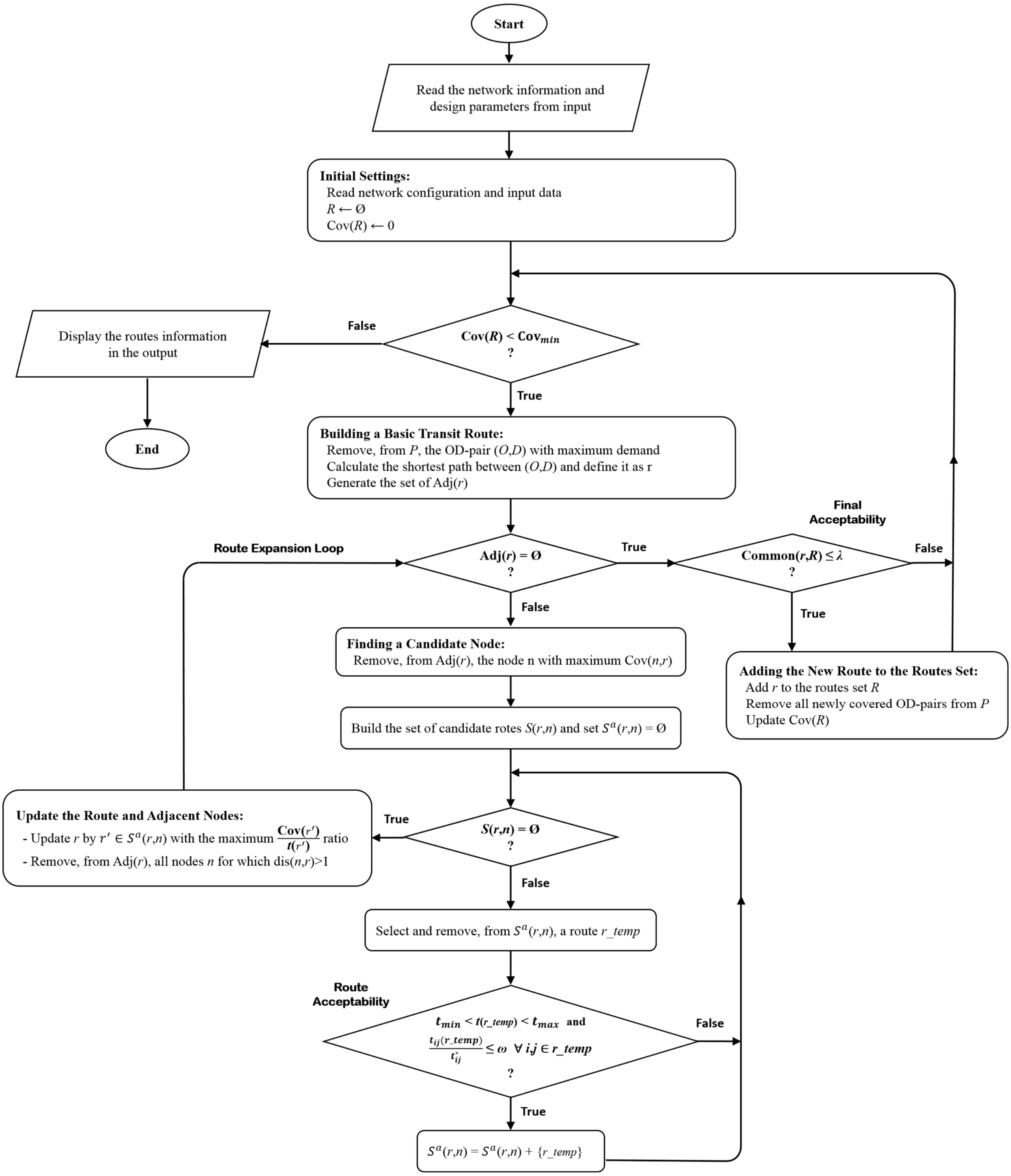
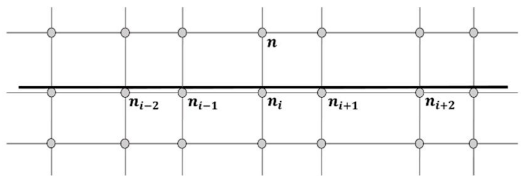
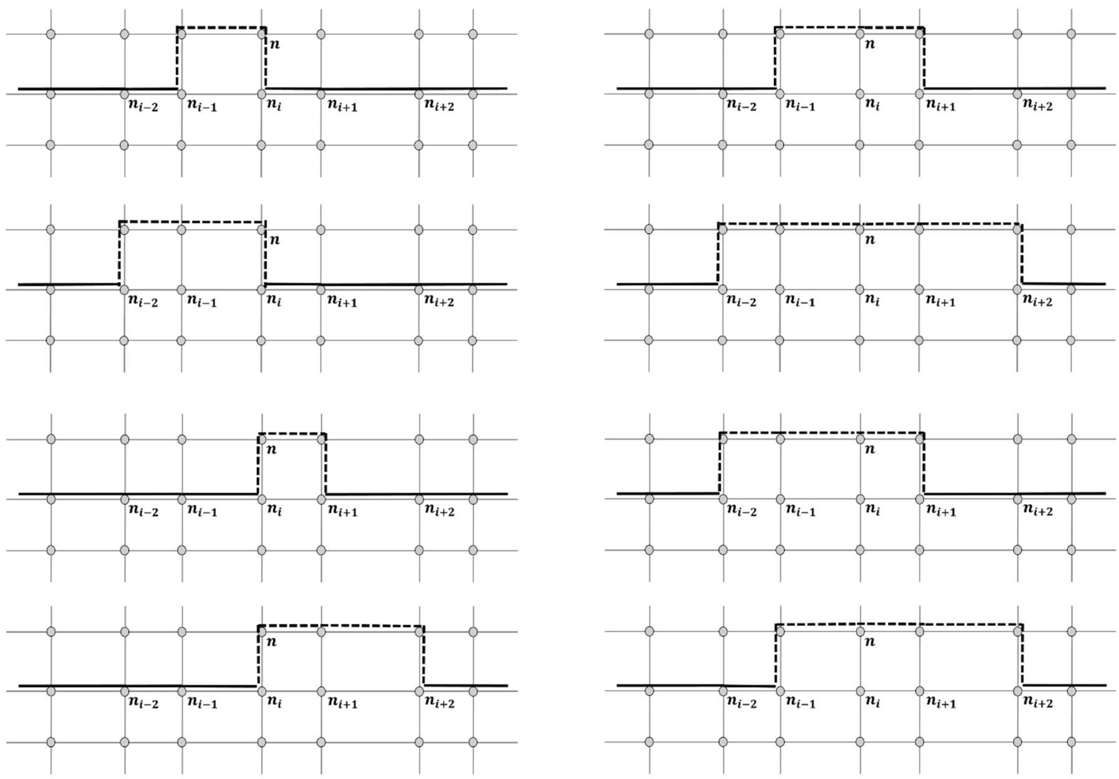


Figure 3. **The proposed constructive algorithm**



(a) The grid network and the transit route



(b) Alternative scenarios for route expansion

The base grid network   The transit route   Potential route expansions

Figure 4. **Alternative route expansions covering node *n* in a grid network**

Therefore, the best route expansion will be determined by the maximum Cov(r_temp)/t(r_temp) ratio for all alternative routes.

After adopting the expanded route, the value of dis(n,r) may turn greater than 1 for some nodes of Adj(r). It is worth to mention that these nodes are removed from Adj(r) after each update. This prevents the algorithm from putting much effort on nodes that are far from the current underlying transit route.

Final Acceptability Condition: To disperse the routes configuration in the network, each generated route r is supposed to have no more than λ common links with the set of previously established routes. This is shown as a final acceptability condition after the route expansion loop. If the newly generated route satisfies this constraint, it will be added to the routes set R. Otherwise, a new iteration is started by the algorithm.

Adding the New Route to the Routes Set: After expanding and finalizing the new transit route, r, the routes set is updated by including r. Also, all node-pairs lying within the route r are removed from P as covered origin-destinations. The coverage of the routes set is finally updated before a new iteration for routes generation is started.

4. Illustrative Example

To explore the functionality of the algorithm, a single iteration of the algorithm over an illustrative example is discussed in this section. A grid network with 12 nodes (3×4 nodes) is considered as depicted in Figure 5. The values written next to the links are considered as their associated travel times in minutes.

Table 1 also illustrates the values of the travel demand from node i to node j, namely D_{ij} ($1 \leq i,j \leq 12$). Since the functionality of the algorithm is based on undirected travel demand between nodes, namely $D(i,j) = D_{ij}+D_{ji}$, we simply consider a symmetric pattern for demand matrix in Table 1. It is assumed that design parameters are T_{min}=15 (minutes), T_{max}=25 (minutes), ω=2, and λ=4.

The algorithm starts with R=Ø at the stage of initial settings. Since demand coverage initially equals to zero, the algorithm goes to the step of building a basic transit route. At this step, a pair of nodes with maximum demand, $D(i,j)$, must be selected. There are three alternatives for this selection: (1,11), (6,7), and (7,11). Let assume that the algorithm picks up the first origin-destination which is (1,11). Then, a shortest-path is built between 1 and 11 as r = {1-5-6-10-11}. This shortest-path (as shown in Figure 6(a)) serves as a basic transit route prior to the route expansion loop. Further, the set of vertices adjacent to route r is built as Adj(r) = {2,7,9,12}.

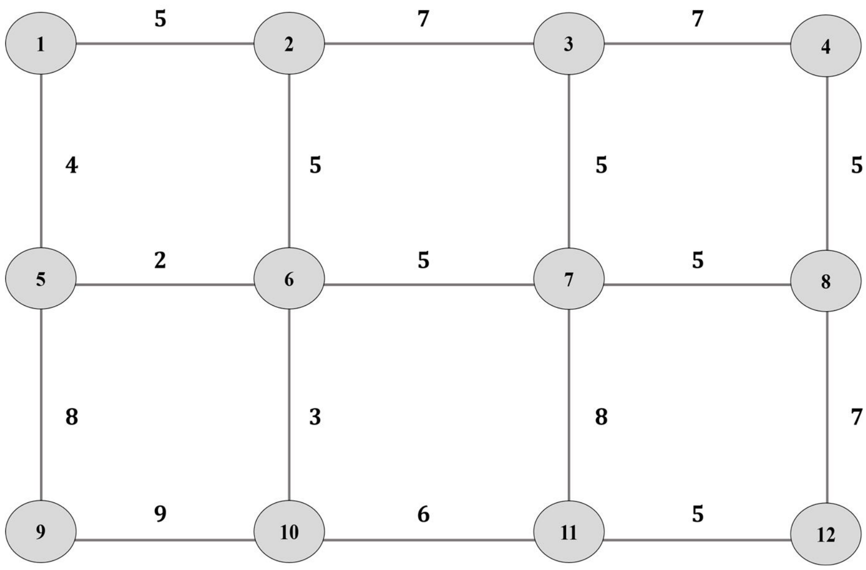


Figure 5. **The 3×4 grid network used as illustrative example**

Table 1. The demand matrix for the illustrative example (in passengers per hour)

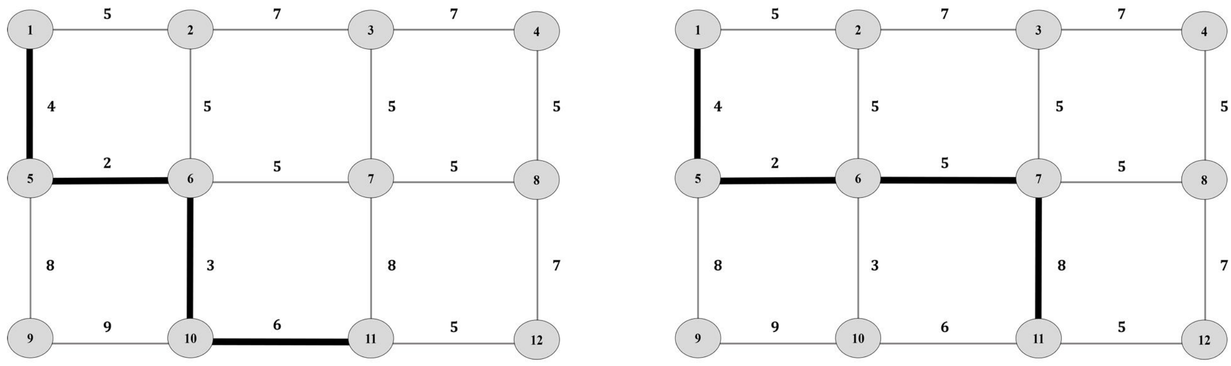
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1	0	1000	1450	0	700	600	0	0	100	0	1500	0
2	1000	0	1200	850	0	900	700	250	150	100	1000	1300
3	1450	1200	0	750	600	0	400	0	0	1210	1300	0
4	0	850	750	0	900	100	100	550	700	0	1200	0
5	700	0	600	900	0	650	0	0	400	1000	700	650
6	600	900	0	100	650	0	1500	1350	0	0	1200	150
7	0	700	400	100	0	1500	0	500	730	690	1500	600
8	0	250	0	550	0	1350	500	0	1490	1470	1150	500
9	100	150	0	700	400	0	730	1490	0	990	800	1200
10	0	100	1210	0	1000	0	690	1470	990	0	0	700
11	1500	1000	1300	1200	700	1200	1500	1150	800	0	0	650
12	0	1300	0	0	650	150	600	500	1200	700	650	0

In the route expansion loop, since the set of Adj(r) is not empty, the algorithm starts by finding the node n with maximum Cov(n,r). The values of Cov(n,r) are 60000, 73800, 45800, and 43000 (passengers per hour) for nodes 2, 7, 9, and 12, respectively. As a result, the node n=7 is picked up as the candidate node to be inserted into the route r. considering either node 6 or 11 as the adjacent node to n=7, there are totally 4 different choices (including the basic route r) for inserting the node 7 into r = {1-5-6-10-11}. These choices are listed in Table 2 as well as the corresponding calculations to select the expanded route. As illustrated in Table 2, the maximum value of Cov(r_exp)/t(r_exp) ratio corresponds to the route expansion r_exp = {1-5-6-7-11}. Therefore, the route r = {1-5-6-10-11} between 1 and 11 is updated to {1-5-6-7-11} as shown in Figure 6 (b).

The set of Adj(r) is also updated by removing node 7. However, the remaining nodes of Adj(r) i.e. 2, 9, and 12 will still stay at 1-link distance from the updated route. These nodes will be kept in Adj(r) for the next iteration of route expansion loop.

Table 2. Alternative choices and calculations for expanding the route r = {1-5-6-10-11} in the illustrative example

Alternative choices for the expanded route, r_temp		t(r_temp) (minutes)	Acceptablity	Cov(r_temp) (passengers per hour)	Cov(r_temp)/t(r_temp) ('passengers per hour' per 'minute')
Basic Transit Route	{1-5-6-10-11}	15	✓	12700	847
Inserting between nodes 6 and 11	{1-5-6-7-11}	19	✓	16700	879
Inserting between nodes 1 and 6	{1-2-3-7-6-10-11}	31	×	-	-
Inserting between nodes 1 and 11	{1-2-6-7-11}	23	✓	19800	861



(a) Before route expansion

(b) After route expansion

Figure 6. The transit route between nodes 1 and 11 before and after expansion

For brevity, the remaining iterations of route expansion loop are not reported here. However, it is worth mentioning that all other route expansion choices during the subsequent iterations will be either infeasible or worse than the expanded route {1-5-6-7-11}. Consequently, the route expansion loop terminates by the route r = {1-5-6-7-11} which leads to the value of 100×Cov(r)/(total travel demand) = 100×16700/80360 = 20.8 (%) as demand coverage. A notable share of the demand (i.e. more than one fifth of the total demand), therefore, gets covered at the first iteration of the algorithm.

5. Numerical Results

This section reports numerical results of the algorithm on a 60-nodes grid network. The general topology of nodes/links is shown in Figure 7. The network is used for illustrative purposes and is not a real one. Following Nie's "VNET" software [Nie, 2016], random link travel-times and demand matrix are adopted for this grid network. The travel demand in the entire network is 333000 passengers per hour. Design parameters are considered to be T_{min}=15, T_{max}=50, ω=2, and λ=12 for this section. Further details, however, are referred to Appendix A at the end of this paper.

Considering 10 (%) as the minimum demand coverage, for example, the algorithm results in 5 transit routes which are illustrated in Table 3. The general configuration of these routes is also sketched in Figure 8. The objective function (i.e. total travel time) and demand coverage associated with these 5 routes are 154.2 (minutes) and 10.2 (%), respectively.

It is interesting also to see how the idea of route expansion can help improving basic shortest-paths between origin-destinations. Two design scenarios are considered to do so. The first scenario involves creating shortest-path routes with no further expansions. All details of this scenario are the same as the algorithm in this paper unless the route expansion loop is omitted. The second scenario is also the algorithm with route expansion as discussed in this paper. Figure 9 shows the results of both scenarios for the 60-nodes grid network.

It can be observed from Figure 9 that, for all levels of demand coverage, the operational costs corresponding to expanded routes are less than those of shortest-path routes. Figure 10 shows the amount of improvements obtained after expansion of the routes, i.e. the vertical gap between the two curves in Figure 9. Note that the values of demand coverage for the curves of Figure 9 are not the same. Therefore, a linear interpolation has been applied in Figure 10 to calculate the gap between these curves.

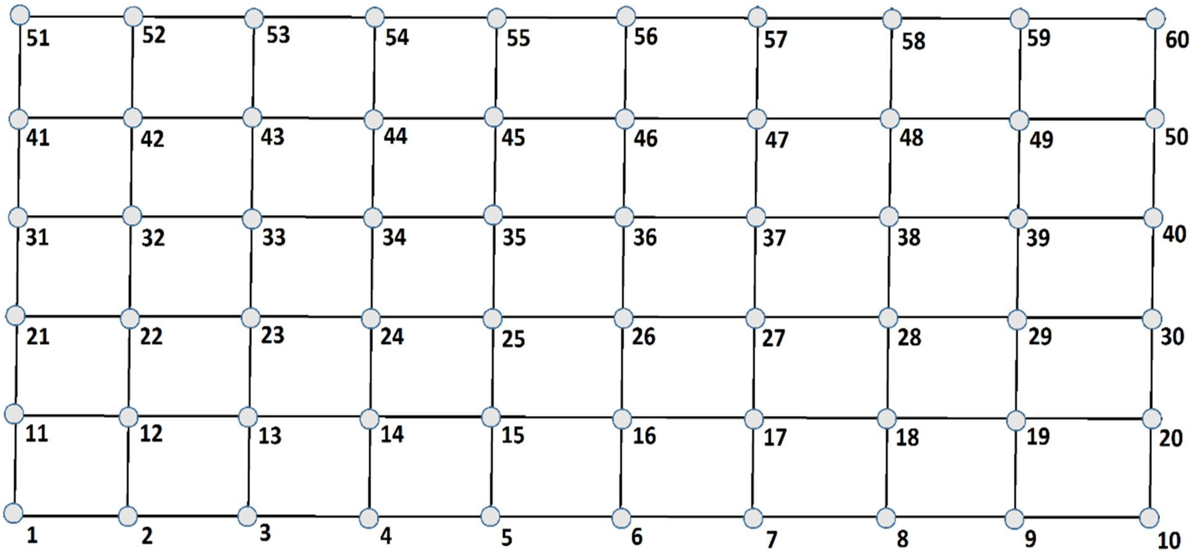


Figure 7. The general topology of the 6×10 grid network

Table 3. The information of 5 routes generated by the algorithm to satisfy demand coverage of 10 (%)

Generated Route, r		T(r) (minutes)	Covered Demand (Passengers per hour)	Cov(r) (%)
$r1$	{47-46-45-44-43-33-23-22-12-2-1}	39.4	11964	3.6
$r2$	{27-17-16-15-14-13-3}	23.2	4578	1.4
$r3$	{54-44-34-24-23-13-3-4}	28.1	5454	1.6
$r4$	{12-22-23-24-25-26-16-6}	29.3	4020	1.2
$r5$	{54-44-34-35-25-15-16-17-18-8}	34.3	7914	2.4

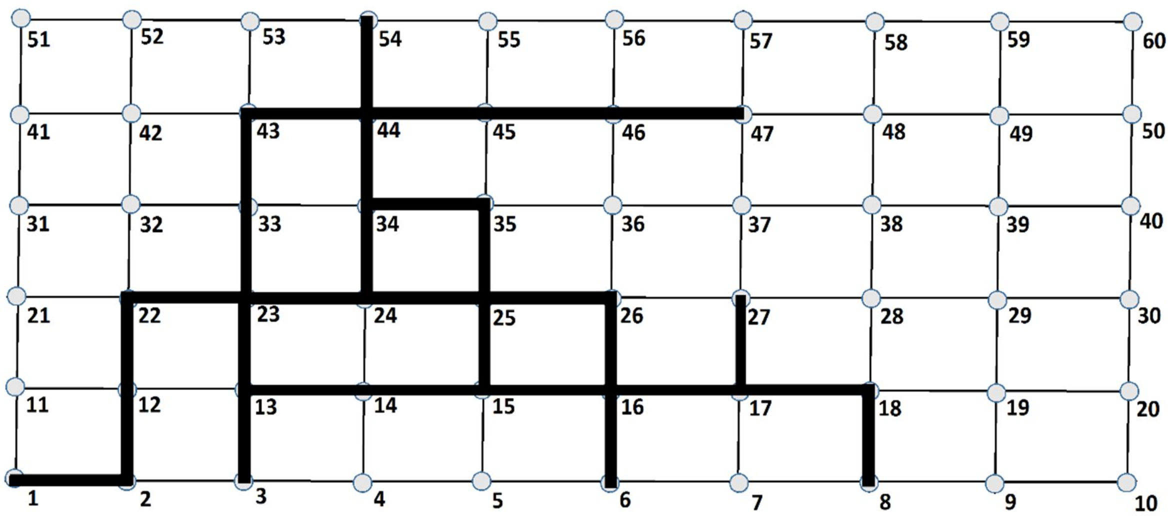


Figure 8. The general configuration of transit routes generated to satisfy demand coverage of 10 (%)

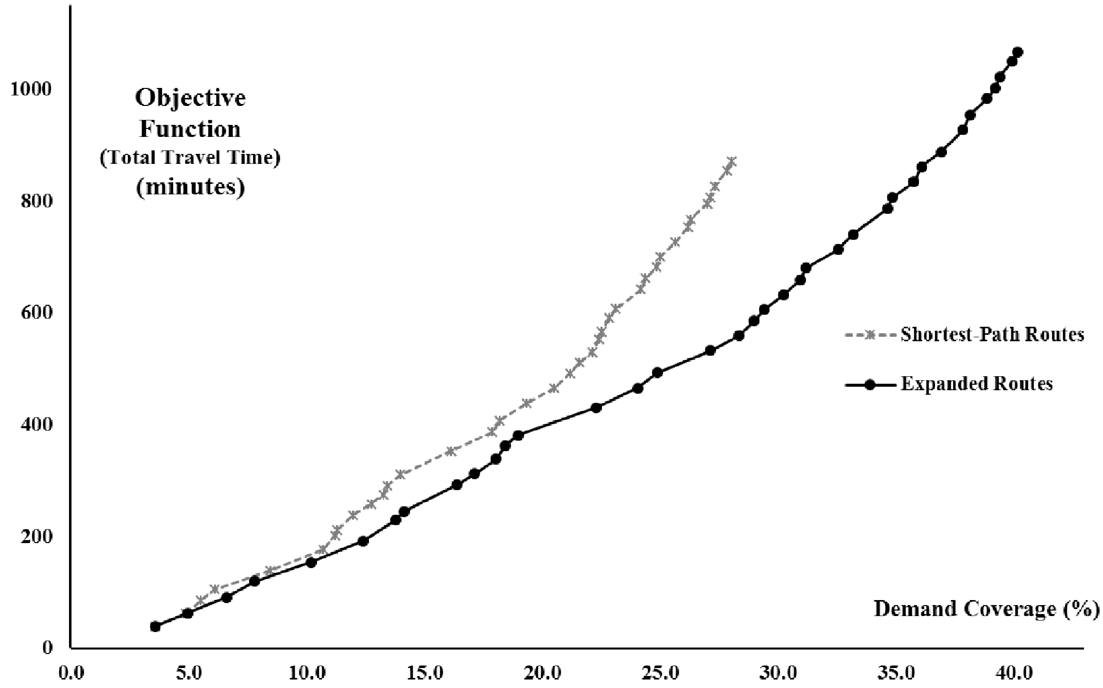


Figure 9. **Route generation results in two scenarios: shortest-path routes versus and expanded routes**

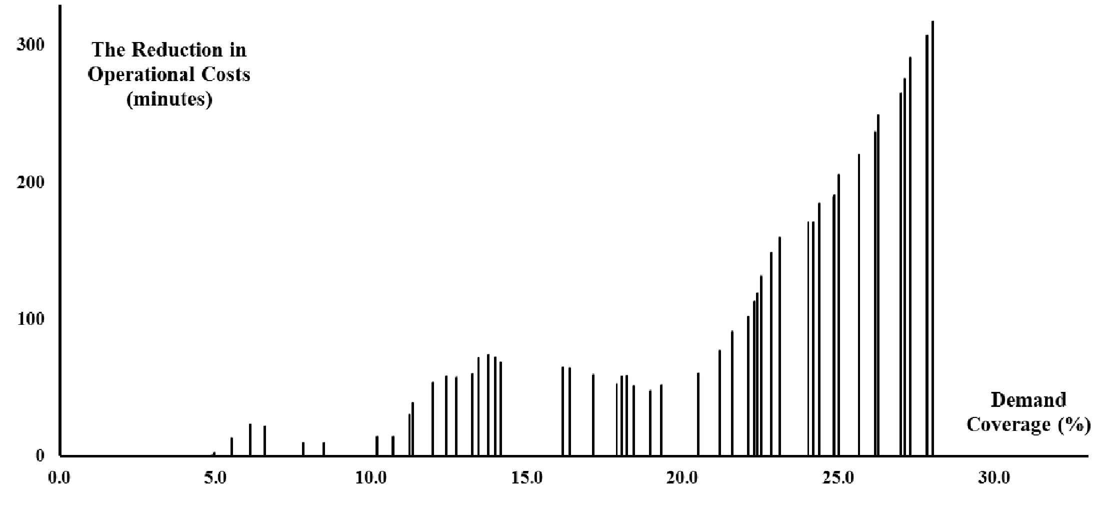


Figure 10. **Objective function improvements obtained by the route expansion algorithm**

The results of the proposed algorithm are in accordance with the previous studies, e.g. Baaj and Mahmassani [1995], in a sense that the objective function increases more sharply as the level of demand coverage is raised. The reason behind this observation is that a main part of the demand matrix gets covered in the first iterations of the algorithm when origin-destinations with the heaviest demand are selected. As a result, covering further origin-destinations will become more costly as the algorithm proceeds.

It is also worthy to note that the gap between the two curves, as depicted in Figure 10, has an overall increasing trend. In other words, applying the idea of expanded routes saves more operational costs in high levels of the demand coverage than it did in low levels. This observation can be justified by the fact that, as the number of routes increases, there is more room for the route expansion algorithm to cover further high-demand nodes along with the routes. However, a more general judgment on this observation would require deeper studies accounting for the demand pattern, length of routes, etc.

6. Concluding Remarks

This paper presented a constructive algorithm with detailed descriptions for designing transit routes in a grid network. Alternative route expansion scenarios were considered to improve the route layout. The results over a 60-nodes random grid network supported that the algorithm is capable of improving (i.e. reducing) the total duration of transit routes in different levels of direct demand coverage.

We applied a grid network with random attributes, in this paper, to report the results. However, to obtain more realistic results and make more general judgments, a research direction is to apply the algorithm on several real transportation networks with grid structure. Design and implementation of constructive algorithms for other special networks, e.g. radial networks, and reporting the results in comparison with this study would further extend the findings of this paper. Also, developing constructive algorithms based on the concepts other than node insertion (e.g. "pair-node insertion" in the study of Mauttone and Urquhart [2009]) and comparing the results with this study may be another interesting topic for future research.

7. Acknowledgement

The authors would like to thank the anonymous referees of the International Journal of Transportation Engineering (IJTE) for many insightful and constructive comments that enhanced the presentation of this paper.

8. References

- Baaj, M. H. and Mahmassani, H. S. (1991) "An AI-based approach for transit route system planning and design", Journal of Advanced Transportation, Vol. 25, No. 2, pp. 187-209.
- Baaj, M.H. and Mahmassani, H.S. (1995) "Hybrid route generation heuristic algorithm for the design of transit networks", Transportation Research C: Emerging Technologies, Vol. 3, No. 1, pp. 31–50.
- Badia, H., Estrada, M. and Robusté, F. (2014) "Competitive transit network design in cities with radial street patterns", Transportation Research Part B: Methodological, Vol. 59, pp. 161-181.
- Cancela, H., Mauttone, A. and Urquhart, M.E. (2015) "Mathematical programming formulations for transit network design", Transportation Research Part B: Methodological, Vol. 77, pp. 17-37.
- Ceder, A. and Wilson, N. H. (1986) "Bus network design", Transportation Research Part B: Methodological, Vol. 20, No. 4, pp. 331-344.
- Farahani, R.Z., Miandoabchi, E., Szeto, W.Y. and Rashidi, H. (2013) "A review of urban transportation network design problems", European Journal of Operational Research, Vol. 229, No. 2, pp. 281-302.
- Guihaire, V. and Hao, J. K. (2008) "Transit network design and scheduling: A global review", Transportation Research Part A: Policy and Practice, Vol. 42, No. 10, pp. 1251-1273.
- Ibarra-Rojas, O.J., Delgado, F., Giesen, R. and Muñoz, J.C. (2015). "Planning, operation, and control of bus transport systems: A literature review", Transportation Research Part B: Methodological, Vol. 77, pp. 38-75.

- Kepaptsoglou, K. and Karlaftis, M. (2009) "Transit route network design problem: review", Journal of Transportation Engineering, Vol. 135, No. 8, pp. 491–505.
- Kermanshahi, S., Shafahi, Y. and Bagherian, M. (2015). "Application of a new rapid transit network design model to bus rapid transit network design: case study Isfahan metropolitan area", Transport, Vol. 30, No. 1, pp. 93-102.
- Mandl, C. E. (1980) "Evaluation and optimization of urban public transportation networks", European Journal of Operational Research, Vol. 5, No. 6, pp. 396-404.
- Mauttone, A. and Urquhart, M. E. (2009) "A route set construction algorithm for the transit network design problem", Computers and Operations Research, Vol. 36, No. 8, pp. 2440-2449.
- Nie, Y. M. (2016) "NUTREND", http://translab.civil.northwestern.edu/nut
- rend/?page_id=53. Accessed 1 February 2016.
- Schöbel, A. (2012) "Line planning in public transportation: models and methods", OR Spectrum, Vol. 34, No. 3, pp. 491-510.
- Snellen, D., Borgers, A. and Timmermans, H. (2002) "Urban form, road network type, and mode choice for frequently conducted activities: a multilevel analysis using quasi-experimental design data", Environment and Planning A, Vol. 34, No. 7, pp. 1207-1220.
- Stern, R. (1996) "Passenger transfer system review", Transportation Research Board, Washington, D.C.
- Zhao, F. and Zeng, X. (2006) "Simulated annealing–genetic algorithm for transit network optimization", Journal of Computing in Civil Engineering, Vol. 20, No. 1, pp. 57-68.

Appendix A. Travel time and demand information for the 6×10 grid network

Given that T(*i-j*) is the travel time of link *i-j* in the network, the travel times (in minutes) associated with 104 links of the 6×10 grid network are as follows:

T(1-3)=3.2, T(1-12)=4.5,

T(2-3)=3.2, T(2-5)=4.6, T(2-14)=5.8,

T(3-5)=4.6, T(3-7)=3.8, T(3-16)=3.7,

T(4-7)=3.8, T(4-9)=4.5, T(4-18)=5.1,

T(5-9)=4.5, T(5-11)=4.0, T(5-20)=4.8,

T(6-11)=4.0, T(6-13)=4.5, T(6-22)=4.3,

T(7-13)=4.5, T(7-15)=5.5, T(7-24)=3.5,

T(8-15)=5.5, T(8-17)=4.1, T(8-26)=3.7,

T(9-17)=4.1, T(9-19)=4.9, T(9-28)=5.1,

T(10-19)=4.9, T(10-30)=6.0,

T(11-12)=4.5, T(11-23)=4.7, T(11-32)=3.8, T(12-14)=5.8, T(12-23)=4.7, T(12-25)=5.3, T(12-34)=3.4,

T(13-16)=3.7, T(13-25)=5.3, T(13-27)=3.6, T(13-36)=4.5,

T(14-18)=5.1, T(14-27)=3.6, T(14-29)=3.8, T(14-38)=5.7,

T(15-20)=4.8, T(15-29)=3.8, T(15-31)=3.2, T(15-40)=3.5,

T(16-22)=4.3, T(16-31)=3.2, T(16-33)=4.1, T(16-42)=5.1,

T(17-24)=3.5, T(17-33)=4.1, T(17-35)=3.1, T(17-44)=4.8,

T(18-26)=3.7, T(18-35)=3.1, T(18-37)=5.4, T(18-46)=4.6,

T(19-28)=5.1, T(19-37)=5.4, T(19-39)=4.4, T(19-48)=3.1,

T(20-30)=6.0, T(20-39)=4.4, T(20-50)=3.2,

T(21-32)=3.8, T(21-43)=5.4, T(21-52)=5.6,

T(22-34)=3.4, T(22-43)=5.4, T(22-45)=3.1, T(22-54)=5.7,

T(23-36)=4.5, T(23-45)=3.1, T(23-47)=4.1, T(23-56)=3.3,

T(24-38)=5.7, T(24-47)=4.1, T(24-49)=6.0, T(24-58)=4.3,

T(25-40)=3.5, T(25-49)=6.0, T(25-51)=3.3, T(25-60)=5.5,

T(26-42)=5.1, T(26-51)=3.3, T(26-53)=5.0,

T(27-44)=4.8, T(27-53)=5.0, T(27-55)=3.3,

T(28-46)=4.6, T(28-55)=3.3, T(28-57)=5.3,

T(29-48)=3.1, T(29-57)=5.3, T(29-59)=3.3,

T(30-50)=3.2, T(30-59)=3.3, T(31-52)=5.6,

T(32-54)=5.7,

T(33-56)=3.3,

T(34-58)=4.3,

and T(35-60)=5.5.

Also, Table *A* illustrates travel demand values (in passengers per hour) between all origin-destinations of the network.

Table. A. Travel demand matrix for the 6×10 grid network

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	177	39	138	18	96	21	90	102	84	162	177	72	111	54	6	87	54	6	57	147	63	144	117	27	78	87	147	33	30
2	177	0	90	108	60	126	153	27	105	9	63	147	9	114	162	6	129	117	168	24	36	33	48	33	48	147	63	33	81	15
3	39	90	0	177	24	150	72	54	24	144	39	18	48	120	150	171	132	24	33	9	63	138	99	63	159	162	183	33	12	42
4	138	108	177	0	132	126	168	144	96	78	54	81	126	15	93	12	87	72	156	30	57	108	18	54	15	69	150	150	90	45
5	18	60	24	132	0	24	156	141	150	117	177	69	81	15	180	102	21	96	90	63	24	138	159	102	15	150	57	180	177	120
6	96	126	150	126	24	0	111	9	117	180	120	183	33	114	99	39	120	63	60	12	174	57	141	159	51	27	162	57	165	45
7	21	153	72	168	156	111	0	156	27	72	54	174	102	78	45	33	117	165	27	135	42	21	36	81	84	66	63	123	168	126
8	90	27	54	144	141	9	156	0	99	129	60	6	111	84	39	129	30	132	126	84	81	108	153	108	126	87	177	150	42	123
9	102	105	24	96	150	117	27	99	0	87	84	90	150	18	9	126	42	27	168	78	15	21	12	30	51	87	171	153	117	60
10	84	9	144	78	117	180	72	129	87	0	165	165	90	78	66	147	123	183	66	21	126	144	9	108	129	75	93	60	6	108
11	162	63	39	54	177	120	54	60	84	165	0	30	171	24	156	99	63	174	42	153	24	132	132	126	105	15	93	18	147	123
12	177	147	18	81	69	183	174	6	90	165	30	0	159	183	54	147	54	147	108	171	84	9	171	156	12	81	183	117	6	123
13	72	9	48	126	81	33	102	111	150	90	171	159	0	132	123	108	78	117	102	153	156	42	105	120	33	183	75	30	141	96
14	111	114	120	15	15	114	78	84	18	78	24	183	132	0	36	96	159	177	72	27	21	126	153	42	111	150	174	150	63	9
15	54	162	150	93	180	99	45	39	9	66	156	54	123	36	0	75	60	24	21	36	147	87	69	168	96	102	21	180	24	141
16	6	6	171	12	102	39	33	129	126	147	99	147	108	96	75	0	141	162	81	147	78	93	60	78	183	150	72	168	168	147
17	87	129	132	87	21	120	117	30	42	123	63	54	78	159	60	141	0	144	120	72	120	69	126	165	180	87	135	36	144	153
18	54	117	24	72	96	63	165	132	27	183	174	147	117	177	24	162	144	0	27	165	114	114	15	108	141	111	138	171	27	72
19	6	168	33	156	90	60	27	126	168	66	42	108	102	72	21	81	120	27	0	117	6	87	63	9	48	183	36	30	177	69
20	57	24	9	30	63	12	135	84	78	21	153	171	153	27	36	147	72	165	117	0	111	57	162	102	138	138	69	12	66	63
21	147	36	63	57	24	174	42	81	15	126	24	84	156	21	147	78	120	114	6	111	0	156	78	159	126	90	123	126	114	132
22	63	33	138	108	138	57	21	108	21	144	132	9	42	126	87	93	69	114	87	57	156	0	96	45	117	114	117	27	108	75
23	144	48	99	18	159	141	36	153	12	9	132	171	105	153	69	60	126	15	63	162	78	96	0	120	159	27	135	45	183	9
24	117	33	63	54	102	159	81	108	30	108	126	156	120	42	168	78	165	108	9	102	159	45	120	0	102	165	126	123	183	87
25	27	48	159	15	15	51	84	126	51	129	105	12	33	111	96	183	180	141	48	138	126	117	159	102	0	33	12	63	87	54
26	78	147	162	69	150	27	66	87	87	75	15	81	183	150	102	150	87	111	183	138	90	114	27	165	33	0	57	96	72	66
27	87	63	183	150	57	162	63	177	171	93	93	183	75	174	21	72	135	138	36	69	123	117	135	126	12	57	0	102	9	15
28	147	33	33	150	180	57	123	150	153	60	18	117	30	150	180	168	36	171	30	12	126	27	45	123	63	96	102	0	135	102
29	33	81	12	90	177	165	168	42	117	6	147	6	141	63	24	168	144	27	177	66	114	108	183	183	87	72	9	135	0	72
30	30	15	42	45	120	45	126	123	60	108	123	123	96	9	141	147	153	72	69	63	132	75	9	87	54	66	15	102	72	0
31	81	117	12	84	63	147	159	171	57	168	21	75	126	33	15	24	21	75	183	108	54	126	117	168	42	15	132	33	126	18
32	60	63	39	129	105	102	174	9	123	144	120	183	177	141	21	90	24	72	18	168	111	180	168	96	12	129	18	99	117	156
33	111	150	45	159	165	51	150	66	30	144	162	153	75	39	96	54	24	39	27	48	105	135	141	30	27	33	84	57	102	171
34	66	135	168	90	15	78	66	60	57	180	108	48	168	12	105	126	171	183	36	6	27	33	108	108	141	24	138	99	87	111
35	39	177	177	168	72	120	153	81	135	153	126	105	177	165	27	87	60	27	63	72	159	36	21	9	147	51	69	150	105	144
36	165	171	174	99	36	153	12	27	171	54	174	129	126	72	162	105	60	156	147	180	180	114	6	51	60	45	135	120	171	102
37	165	156	111	81	105	102	135	72	48	126	12	75	144	66	105	63	183	9	54	96	72	45	75	150	147	123	69	66	15	159
38	30	108	117	129	78	30	60	132	42	138	75	9	123	120	135	150	126	54	123	141	57	153	18	168	183	81	51	51	66	147
39	42	108	144	162	18	159	171	102	15	165	33	147	60	123	162	30	162	159	138	126	171	75	84	45	147	162	135	27	21	21
40	96	81	123	135	123	93	21	150	75	72	60	78	96	54	90	9	30	27	96	174	69	174	60	45	24	114	105	114	147	9
41	171	174	63	51	84	138	21	144	54	27	174	153	27	135	48	24	45	153	39	162	141	162	9	48	78	78	48	162	21	123
42	90	84	144	24	129	15	108	72	69	135	129	147	111	87	39	33	162	81	78	96	105	81	147	81	45	123	171	42	147	75
43	147	30	60	141	57	120	135	132	123	66	138	144	126	39	60	60	183	147	132	156	69	153	144	120	9	183	144	63	12	87
44	171	87	84	18	96	72	66	6	144	180	69	84	162	21	177	99	63	93	132	39	105	117	33	177	42	162	45	87	168	111
45	153	165	108	12	177	165	33	123	15	162	108	60	24	15	180	21	123	72	105	156	147	135	108	75	162	138	18	12	159	111
46	120	135	114	180	51	54	105	177	51	102	108	93	90	57	36	45	93	123	12	177	84	81	138	48	138	57	15	114	72	153
47	183	138	42	117	66	117	96	147	96	33	96	153	81	33	126	54	72	75	27	177	177	6	96	39	159	159	108	63	75	141
48	21	96	144	96	15	126	78	6	126	87	177	75	51	42	120	69	30	45	114	84	90	165	6	108	15	57	93	36	93	117
49	24	84	156	117	153	18	180	135	177	66	111	90	153	18	87	42	129	135	18	75	171	84	48	45	54	150	84	168	84	78
50	117	60	39	141	114	135	111	135	168	27	141	21	126	168	33	174	174	84	48	18	159	111	108	69	159	18	42	168	39	27
51	69	39	108	51	75	63	54	6	60	123	96	162	18	177	27	111	72	75	147	102	87	6	63	54	78	45	129	69	39	27
52	180	27	72	54	27	75	60	159	147	117	105	48	48	60	45	33	165	144	162	150	21	33	9	63	42	99	12	33	114	108
53	54	18	30	63	105	36	105	123	102	162	18	9	123	36	93	111	60	69	180	24	165	156	27	33	141	36	111	87	144	144
54	177	81	33	183	129	129	75	183	168	30	84	78	174	75	147	6	99	132	45	57	150	165	150	123	183	126	42	30	42	21
55	114	51	108	90	162	126	114	162	150	111	39	171	123	129	150	72	48	63	180	48	75	9	144	84	27	183	126	84	33	60
56	123	177	165	171	132	18	51	66	165	135	51	27	141	150	51	147	81	165	144	51	42	72	147	48	144	54	99	51	117	78
57	18	93	36	21	42	105	21	18	174	96	183	135	132	57	174	78	21	12	75	111	6	60	66	129	66	27	99	168	54	99
58	75	141	174	174	60	120	171	120	39	159	180	63	51	108	135	117	9	57	27	30	135	87	102	57	156	33	9	111	120	93
59	24	81	33	15	141	159	147	57	159	135	15	126	18	21	96	135	57	183	108	174	48	156	87	15	18	147	24	183	93	138
60	111	126	57	102	171	132	144	108	177	9	99	54	39	36	39	57	72	120	72	87	141	51	126	12	87	147	105	114	141	126

Table. A. Travel demand matrix for the 6×10 grid network (continued)

	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
1	81	60	111	66	39	165	165	30	42	96	171	90	147	171	153	120	183	21	24	117	69	180	54	177	114	123	18	75	24	111
2	117	63	150	135	177	171	156	108	108	81	174	84	30	87	165	135	138	96	84	60	39	27	18	81	51	177	93	141	81	126
3	12	39	45	168	177	174	111	117	144	123	63	144	60	84	108	114	42	144	156	39	108	72	30	33	108	165	36	174	33	57
4	84	129	159	90	168	99	81	129	162	135	51	24	141	18	12	180	117	96	117	141	51	54	63	183	90	171	21	174	15	102
5	63	105	165	15	72	36	105	78	18	123	84	129	57	96	177	51	66	15	153	114	75	27	105	129	162	132	42	60	141	171
6	147	102	51	78	120	153	102	30	159	93	138	15	120	72	165	54	117	126	18	135	63	75	36	129	126	18	105	120	159	132
7	159	174	150	66	153	12	135	60	171	21	21	108	135	66	33	105	96	78	180	111	54	60	105	75	114	51	21	171	147	144
8	171	9	66	60	81	27	72	132	102	150	144	72	132	6	123	177	147	6	135	135	6	159	123	183	162	66	18	120	57	108
9	57	123	30	57	135	171	48	42	15	75	54	69	123	144	15	51	96	126	177	168	60	147	102	168	150	165	174	39	159	177
10	168	144	144	180	153	54	126	138	165	72	27	135	66	180	162	102	33	87	66	27	123	117	162	30	111	135	96	159	135	9
11	21	120	162	108	126	174	12	75	33	60	174	129	138	69	108	108	96	177	111	141	96	105	18	84	39	51	183	180	15	99
12	75	183	153	48	105	129	75	9	147	78	153	147	144	84	60	93	153	75	90	21	162	48	9	78	171	27	135	63	126	54
13	126	177	75	168	177	126	144	123	60	96	27	111	126	162	24	90	81	51	153	126	18	48	123	174	123	141	132	51	18	39
14	33	141	39	12	165	72	66	120	123	54	135	87	39	21	15	57	33	42	18	168	177	60	36	75	129	150	57	108	21	36
15	15	21	96	105	27	162	105	135	162	90	48	39	60	177	180	36	126	120	87	33	27	45	93	147	150	51	174	135	96	39
16	24	90	54	126	87	105	63	150	30	9	24	33	60	99	21	45	54	69	42	174	111	33	111	6	72	147	78	117	135	57
17	21	24	24	171	60	60	183	126	162	30	45	162	183	63	123	93	72	30	129	174	72	165	60	99	48	81	21	9	57	72
18	75	72	39	183	27	156	9	54	159	27	153	81	147	93	72	123	75	45	135	84	75	144	69	132	63	165	12	57	183	120
19	183	18	27	36	63	147	54	123	138	96	39	78	132	132	105	12	27	114	18	48	147	162	180	45	180	144	75	27	108	72
20	108	168	48	6	72	180	96	141	126	174	162	96	156	39	156	177	177	84	75	18	102	150	24	57	48	51	111	30	174	87
21	54	111	105	27	159	180	72	57	171	69	141	105	69	105	147	84	177	90	171	159	87	21	165	150	75	42	6	135	48	141
22	126	180	135	33	36	114	45	153	75	174	162	81	153	117	135	81	6	165	84	111	6	33	156	165	9	72	60	87	156	51
23	117	168	141	108	21	6	75	18	84	60	9	147	144	33	108	138	96	6	48	108	63	9	27	150	144	147	66	102	87	126
24	168	96	30	108	9	51	150	168	45	45	48	81	120	177	75	48	39	108	45	69	54	63	33	123	84	48	129	57	15	12
25	42	12	27	141	147	60	147	183	147	24	78	45	9	42	162	138	159	15	54	159	78	42	141	183	27	144	66	156	18	87
26	15	129	33	24	51	45	123	81	162	114	78	123	183	162	138	57	159	57	150	18	45	99	36	126	183	54	27	33	147	147
27	132	18	84	138	69	135	69	51	135	105	48	171	144	45	18	15	108	93	84	42	129	12	111	42	126	99	99	9	24	105
28	33	99	57	99	150	120	66	51	27	114	162	42	63	87	12	114	63	36	168	168	69	33	87	30	84	51	168	111	183	114
29	126	117	102	87	105	171	15	66	21	147	21	147	12	168	159	72	75	93	84	39	39	114	144	42	33	117	54	120	93	141
30	18	156	171	111	144	102	159	147	21	9	123	75	87	111	111	153	141	117	78	27	27	108	144	21	60	78	99	93	138	126
31	0	117	51	180	93	84	18	120	39	183	48	96	108	12	123	15	126	135	111	144	123	102	105	117	102	63	147	183	69	135
32	117	0	111	69	84	120	129	138	84	21	24	168	99	54	111	162	27	183	84	132	60	177	162	126	81	60	60	96	18	69
33	51	111	0	54	66	45	108	168	63	168	114	78	51	51	51	18	117	78	18	90	18	117	45	63	165	87	6	72	153	165
34	180	69	54	0	15	12	81	183	120	48	180	120	150	42	126	156	129	165	84	69	120	114	150	36	33	153	138	102	48	180
35	93	84	66	15	0	42	93	180	138	150	138	78	141	126	165	129	141	39	168	30	156	12	30	168	96	84	57	18	15	6
36	84	120	45	12	42	0	126	90	63	114	24	36	132	105	72	114	45	126	96	24	24	45	108	24	135	177	171	153	15	102
37	18	129	108	81	93	126	0	90	153	54	36	159	153	42	60	96	45	93	33	90	66	69	75	180	135	84	6	51	132	132
38	120	138	168	183	180	90	90	0	51	123	63	165	63	183	33	48	6	90	39	171	42	138	123	42	150	18	90	36	81	171
39	39	84	63	120	138	63	153	51	0	165	21	96	93	9	126	132	111	75	99	45	21	99	102	168	129	39	30	90	129	12
40	183	21	168	48	150	114	54	123	165	0	141	45	180	123	99	102	135	30	168	21	51	132	72	126	102	105	57	45	156	168
41	48	24	114	180	138	24	36	63	21	141	0	171	174	21	150	162	159	153	12	96	42	93	33	45	27	174	102	60	72	108
42	96	168	78	120	78	36	159	165	96	45	171	0	33	72	96	30	18	24	33	60	114	147	45	33	84	114	66	93	99	30
43	108	99	51	150	141	132	153	63	93	180	174	33	0	63	39	162	81	150	33	144	165	42	60	135	93	150	150	156	138	60
44	12	54	51	42	126	105	42	183	9	123	21	72	63	0	144	180	150	153	93	6	132	174	24	21	129	105	99	153	75	48
45	123	111	51	126	165	72	60	33	126	99	150	96	39	144	0	72	96	54	72	51	159	123	66	105	99	102	39	15	141	12
46	15	162	18	156	129	114	96	48	132	102	162	30	162	180	72	0	78	21	144	123	156	48	72	99	45	6	162	30	90	39
47	126	27	117	129	141	45	45	6	111	135	159	18	81	150	96	78	0	105	30	39	9	54	57	45	90	57	15	51	162	57
48	135	183	78	165	39	126	93	90	75	30	153	24	150	153	54	21	105	0	126	183	63	21	174	126	12	123	102	165	54	183
49	111	84	18	84	168	96	33	39	99	168	12	33	33	93	72	144	30	126	0	141	15	126	180	87	150	162	117	48	132	180
50	144	132	90	69	30	24	90	171	45	21	96	60	144	6	51	123	39	183	141	0	126	171	168	93	39	132	162	72	27	96
51	123	60	18	120	156	24	66	42	21	51	42	114	165	132	159	156	9	63	15	126	0	108	51	90	99	138	141	60	66	111
52	102	177	117	114	12	45	69	138	99	132	93	147	42	174	123	48	54	21	126	171	108	0	18	144	135	141	72	81	99	135
53	105	162	45	150	30	108	75	123	102	72	33	45	60	24	66	72	57	174	180	168	51	18	0	69	15	15	156	27	63	183
54	117	126	63	36	168	24	180	42	168	126	45	33	135	21	105	99	45	126	87	93	90	144	69	0	81	144	93	144	129	177
55	102	81	165	33	96	135	135	150	129	102	27	84	93	129	99	45	90	12	150	39	99	135	15	81	0	6	27	42	159	6
56	63	60	87	153	84	177	84	18	39	105	174	114	150	105	102	6	57	123	162	132	138	141	15	144	6	0	87	147	168	165
57	147	60	6	138	57	171	6	90	30	57	102	66	150	99	39	162	15	102	117	162	141	72	156	93	27	87	0	159	162	48
58	183	96	72	102	18	153	51	36	90	45	60	93	156	153	15	30	51	165	48	72	60	81	27	144	42	147	159	0	144	21
59	69	18	153	48	15	15	132	81	129	156	72	99	138	75	141	90	162	54	132	27	66	99	63	129	159	168	162	144	0	165
60	135	69	165	180	6	102	132	171	12	168	108	30	60	48	12	39	57	183	180	96	111	135	183	177	6	165	48	21	165	0