Mixed Network Design Using Hybrid Scatter Search

Zohreh Khooban

System Planning Department, Farab Co. (Energy and Water Projects), Tehran, Iran

Reza Zanjirani Farahani

Operations Research and Management Science Research Group, Department of Management, Kingston Business School, Kingston University, UK

Elnaz Miandoabchi

Logistics and Infrastructures Research Group, Institute for Trade Studies and Research, Tehran, Iran

W.Y. Szeto

Department of Civil Engineering, The University of Hong Kong, Hong Kong

Abstract

This research proposes a bi-level model for the mixed network design problem (MNDP). The upper level problem involves redesigning the current road links' directions, expanding their capacity, and determining signal settings at intersections to optimize the reserve capacity of the whole system. The lower level problem is the user equilibrium traffic assignment problem. By proving that the optimal arc flow solution of the bi-level problem must exist in the boundary of capacity constraints, an exact line search method called golden section search is embedded in a scatter search method for solving this complicated MNDP. The algorithm is then applied to some real cases and finally, some conclusions are drawn on the model's efficiency.

Keywords: Transportation network design; Bi-level programming; User equilibrium traffic assignment; Golden section; Scatter search.

1. INTRODUCTION

The Urban Network Design Problem (UNDP) is a classic decision problem in transportation planning and is concerned with the improvement of urban transportation network systems in order to respond to the growth of travel demand. Nowadays, studying urban transportation network systems is crucial because the speed of the increase in urban transportation demand is higher than that in expanding the transportation system, so this system could not accommodate the increase in demand, while resources available for expanding the system capacity remain limited (Yang and Bell, 1998a). Until now, most UNDPs have been formulated as bi-level problems which in the upper level problem, several investment decisions are made by system owners or planners to optimize the desired objective function.

When it comes to decision variables in the upper level problem, UNDPs are divided into three different classes. The first class is known as the discrete network design problem (DNDP) which only involves discrete decisions (e.g., Long et al., 2010, 2014; Miandoabchi et al., 2012a,b; 2015; Szeto et al., 2014). Typical discrete decisions in the DNDP are constructing new streets, adding new lanes to the existing streets, determining the street directions and their lane allocations, and designing the turning restrictions at intersections. The second type is the continuous network design problem (CNDP) (e.g., Szeto and Lo, 2006; 2008; Lo and Szeto, 2009; Sun et al., 2014; Jiang and Szeto, 2015; Szeto et al., 2015) which deals only with continuous variables such as signal setting of intersections, determining road tolls, and street capacity expansion. The last type is named the mixed network design problem (MNDP) which involves both continuous and discrete variables. There are few research papers in this category. Some recent related researches are Cantarella et al. (2006), Dimitriou et al. (2008), Zhang and Gao (2009), and Gallo et al. (2010). The problem in this research is a kind of MNDP because several discrete and continuous variables are involved.

According to Magnanti and Wong (1984), the decisions in UNDPs can be grouped into strategic, tactical, and operational types, each of which deals with long-term, mid-term, and short-term network design issues, respectively. This paper investigates the strategic decision of street capacity expansion, the tactical decision of one-way two-way streets configuration, and the operational decision of signal setting at intersections. After that, a number of comprehensive reviews have been published by Friesz (1985), Migdalas (1995), and Yang and Bell (1998a) which focus specifically on UNDP. Recently, Farahani et al. (2013) also conducted a comprehensive review on UNDPs' definitions, classifications, objectives, constraints, and solution methods, objectives, constraints, and solution methods, which encompass both road and public transit network design problem.

The street orientation was first considered by Lee and Yang (1994) as the sole network design decision in a bi-level model to maximize the total travel time of the network. After that, in some research, the single level modeling approach along with all or nothing traffic assignment was used for optimizing the street orientations (e.g., Drezner and Wesolowsky, 1997; Drezner and Salhi, 2000; 2002; Drezner and Wesolowsky, 2003). All the other related studies adopted bi-level models with the user equilibrium traffic assignment approach for optimizing the street orientations and other discrete or continuous decisions (e.g., Cantarella et al., 2006; Gallo et al., 2010; Miandoabchi and Farahani, 2011; Miandoabchi et al., 2013).

Street capacity expansion can be considered as the most prevalent decision in UNDP studies. Although this has been modeled in most of the previous research as a continuous variable to simplify the solution method for solving the problem, it has been modeled as a discrete variable in a number of other studies. For example, Steenbrink (1974), LeBlanc (1975), Poorzahedy and Turnquist (1982), Yang and Bell (1998b), Poorzahedy and Abulghasemi (2005), Poorzahedy and Rouhani (2007), Szeto et al. (2010), Miandoabchi and Farahani (2011), and Miandoabchi et al. (2013) have investigated discrete capacity expansion in DNDPs. In MNDPs, only Dimitriou et al. (2008) have modeled this as a discrete variable.

The most common objective function among UNDPs is the minimization of total travel time or cost across the network. Other used objective functions include consumer/social surplus, total distance traveled, minimum construction or construction and travel cost, reserve capacity, etc. In this research, the maximization of reserve capacity is adopted as the objective function. Webster and Cobbe (1966), Allsop (1972), and Wong (1996) investigated this concept for network intersections. However, using this concept as the objective function of the UNDPs was only suggested in the study by Yang and Bell (1998a). Yang and Bell (1998b) introduced a paradox related to network design problems and demonstrated that using the concept of reserve capacity into a network design problem is the best way to avoid this paradox. They also mentioned several advantages of the capacity-based formulation for UNDPs such as formulation simplicity. There are many alternative factors to measure the reserve capacity of a system, but the common one is the multiplier of the origin-destination (O-D) demand matrix of network. In this way, reserve capacity can be defined as the largest multiplier which can be applied to the existing travel demand matrix of the concerned network, such that the street flow capacities are not violated. Reserve capacity is often captured in the CNDPs, while in DNDP only Gao et al. (2005), Miandoabchi and Farahani (2011), and Miandoabchi et al. (2013) have exploited this as the objective function. However, there is no research on using this as the objective function in MNDPs.

In this paper, an MNDP is introduced to maximize the reserve capacity of the whole network. The problem involves two types of discrete variables, namely i) capacity expansion and orientation of the existing streets, and ii) one type of continuous variable, i.e., signal setting. The common approach of bi-level programming is used to model the proposed problem, in which the simple deterministic user equilibrium assignment problem is used in the lower level problem. Table 1 demonstrates a summary on the related studies and compares the main attributes of the problem addressed in this research with them.

Table 1. A summary of the related studies in NDPs.

Reference	Objective	Traffic Assignment	Demand	Decision						Solution Method
				Discrete				Continuous		
				Making Some Streets One-Way	Street Capacity Expansion	Constructing New Streets	Orienting Sequences of Streets	Traffic Light Setting	Street Capacity Expansion	
Wong and Yang (1997)	Max. reserve capacity	DUE	F						●	Sensitivity analysis-based algorithm
Yang and Bell (1998a)	General weighted sum multi-objective	DUE	F			●			●	Enumeration scheme with other methods
Yang and Bell (1998b)	Max. reserve capacity	DUE	F						●	-
Ziyou and Yifan (2002)	Max. reserve capacity	DUE	F					●	●	Sensitivity analysis-based method
Cantarella et al. (2006)	Min. total travel time	DUE	F				●	●		Hill climbing, simulated annealing, tabu search, genetic algorithm, hybrids of tabu search
Zhang and Gao (2009)	Min. total travel cost + construction cost	DUE	F			●			●	Gradient-based method with penalty function
Miandoabchi and Farahani (2011)	Max. reserve capacity	DUE	F	●	●					Hybrid genetic algorithm and an evolutionary simulated annealing
Miandoabchi et al. (2013)	Max. reserve capacity + Min. two travel time related objective functions	DUE	F	●	●	●				Multi-objective algorithms: Hybrid genetic algorithm, evolutionary simulated annealing, and artificial bee colony
Gallo et al. (2010)	Min. Total travel time	SUE	F	●				●		Scatter search algorithm
This Research	Max. reserve capacity	DUE	F	●	●			●		Hybrid scatter search algorithm

In this paper, a hybrid scatter search (HSS) algorithm is developed to solve the proposed problem because of the complexity and non-convexity of MNDPs. In the literature, Gallo et al. (2010) used scatter search to solve their problem, but there are some significant differences between these two works: firstly, our model is more complicated because of adding street capacity expansion as a discrete variable and also incorporating the reserve capacity concept in the problem; besides, the proposed model in Gallo et al. (2010) did not consider capacity constraints which made their solution procedure easier. We propose a hybrid scatter search embedding a golden section search method in the scatter search algorithm to cope with capacity constraints in MNDPs. Our proposed algorithm can be an appropriate replacement for the sensitivity analysis based (SAB) algorithm, which is a very common procedure in solving MNDPs and may not be able to solve problems with specific network settings, due to the non-existence of their matrix inversions (Miandoabchi and Farahani, 2011).

The contributions of this paper include the following. First, a new and more complicated MNDP is proposed. Second, a new hybrid scatter search algorithm is developed to solve the problem. The reminder of this paper is organized as follows. The notations and mathematical model of the problem are defined in Section 2. The solution algorithm is presented in Section 3. In Section 4, numerical examples for several real networks are given. Finally, conclusions are drawn in Section 5.

2. OPTIMIZATION MODEL

In this section, a bi-level mathematical model is formulated for our proposed problem, in which both levels are formulated as non-linear constrained optimization models. Three groups of variables are optimized in the upper level. In this level, network policy makers create a configuration for the network by adjusting two groups of median term variables consisting of redesigning directions and expansion of current link capacities. Besides, signal setting variables are also adjusted as a group of short-term decision variables. In an overall view, when a group of variables are characterized in their own levels, they will be sent to another level as inputs. Figure 1 depicts the bi-level nature of the problem.

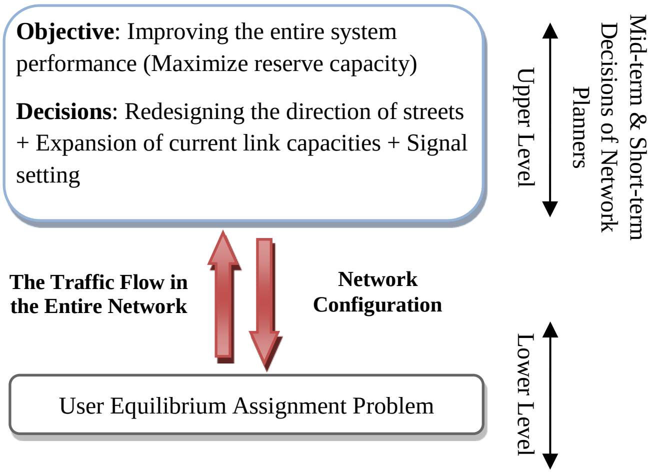

Figure 1. Data transfer between the two levels.

In the studied problem, a basic network with known street (link) capacities, directions, and intersections to be improved exists in advance. Moreover, travel demand between each node pair is known and fixed. Additionally, network users follow the user equilibrium principle. The input data for this problem are as follows:

- Current urban transportation network graph (including link capacities and directions);
- Current O-D matrix;
- The capacities, lengths, and free flow travel times of links;
- The unit cost of widening a link (adding a lane) and maximum budget for expanding link capacity;
- The maximum number of total possible changes in link directions;

- Upper and lower bounds of the signal setting variable and the lower bound of the reserve capacity (the matrix multiplier).

This research intends to make the following decisions for the problem:

- The configuration of one-way or two-ways links;
- The reserve capacity (matrix multiplier) of the improved network;
- The capacity increment of each link;
- Signal setting of each intersection;
- Equilibrium flows on network links.

The upper level objective function is the O-D matrix multiplier. As mentioned before, it is the first time to consider concept of reserve capacity for a MNDP. Table 2 consists of key notations used in this bi-level mathematical model for this specified MNDP. The mathematical model developed for this problem is based on the bi-level programming approach used in UNDPs. This model represents the leader-follower or Stackelberg game, in which the network authority as the leader decides the network design and the network users as the followers who react to the design scenarios by changing their routes.

Table 2. List of key notations used in problem formulation.

Notation	Description
Sets	
S	Set of all links s
A	Set of all arcs (i, j)
A_s	Set of arcs (i, j) and (j, i) belonging to the link s
E	Set of signal-controlled intersections in the network
W	Set of all pairs of origin and destination nodes (m, n)
R	Set of all paths in the network
R_{mn}	Set of paths r between origin node m and destination node n for all $(m,n) \in W$, $R_{mn} \in R$
Decision variables	
y_s	Variable representing the direction of link s. It takes values of -1, 1, and 2.
u_s	Binary variable of link s for all $s \in S$ which equals 1 if link s is selected for capacity expansion and 0 otherwise.
δ_{mn}^{ij}	Binary variable, which equals 1 if arc (i, j) is built on paths between m and n, (m, n) $\in W$ and 0 otherwise.
μ	O-D matrix multiplier
λ_{ij}	Signal setting variable (proportion of green time) for arc (i, j)
F_r	Flow of path r
Parameters	
θ	Cost of widening per unit length of a link (adding a lane)
γ	Multiplier for increasing the capacity (i.e., width) of a link

η	Maximum available budget for increasing link capacities
q_{mn}	Total demand from origin node *m* to destination node *n*
d_s	Length of link *s*
t_{ij}^0	Free flow travel time on arc (*i*, *j*)
y_s^0	Initial direction of link *s*
ϕ_{ij}^r	Indicator variable that equals 1 if arc (*i*, *j*) is on path *r* and is 0 otherwise
$\lambda_{\min}$	Lower bound of the proportions of green time
$\lambda_{\max}$	Upper bound of the proportions of green time
μ_0	Lower bound of the O-D matrix multiplier
c_{ij}	Saturation flow of arc (*i*, *j*)
p_{ij}	Maximum acceptable degree of saturation for the flow on arc (*i*, *j*) which takes a value in [0,1] and almost near to 1.
ϑ_{ij}^e	Indicator variable that equals 1 if arc (*i*, *j*) goes into signal-control intersection *e* and is 0 otherwise
Functions	
t_{ij}	Travel time on arc (*i*, *j*)
f_{ij}	Flow on arc (*i*, *j*) that is a function of upper-level decision variables *y*; μ; *u*; λ
C_{ij}	Capacity of arc (*i*, *j*)
Z_I	Upper level objective function
Z_{II}	Lower level objective function

2.1. Upper level formulation

The upper level optimization model is as follows:

$$Maximize\;\; Z_I = \mu\,(f_{ij}, C_{ij}(\lambda_{ij}, u_s, y_s)) \tag{1}$$

Subject to:

$$\frac{(y_s)^2 - y_s}{2} - \delta_{mn}^{ji} \geq 0 \qquad \forall s \in S, (i,j) \in A_s, (m,n) \in W \tag{2}$$

$$\frac{y_s}{|2y_s|} + \frac{1}{2} - \delta_{mn}^{ij} \geq 0 \qquad \forall s \in S, (i,j) \in A_s, (m,n) \in W \tag{3}$$

$$\delta_{mn}^{ij} + \delta_{mn}^{ji} \geq 0 \qquad \forall\, (i,j) \in A_s, (m,n) \in W \tag{4}$$

$$\sum_{s \in S} \theta\,\gamma\,(u_s\, d_s) \leq \eta \tag{5}$$

$$f_{ij}(\lambda_{ij}, u_s) \leq p_{ij} C_{ij}(\lambda_{ij}, u_s, y_s) \qquad \forall s \in S, (i,j) \in A_s \tag{6}$$

$$f_{ji}(\lambda_{ji}, u_s) \leq p_{ji} C_{ji}(\lambda_{ji}, u_s, y_s) \qquad \forall s \in S, (i,j) \in A_s \tag{7}$$

$$C_{ij}(\lambda_{ij}, u_s, y_s) = \left(1 - \frac{(y_s)^2 - y_s}{2|y_s|}\right)\lambda_{ij}(c_{ij}(1 + \gamma\, u_s)) \qquad \forall s \in S, (i,j) \in A_s \tag{8}$$

$$C_{ji}(\lambda_{ji}, u_s, y_s) = \left(\frac{(y_s)^2 - y_s}{2|y_s|}\right)\lambda_{ji}(c_{ji}(1+\gamma u_s)) \qquad \forall s \in S, (i,j) \in A_s \tag{9}$$

$$\sum_{(i,j)\in A} \lambda_{ij} . \vartheta_{ij}^{e} = 1 \qquad \forall e \in E \tag{10}$$

$$\lambda_{\min} \le \lambda_{ij} \le \lambda_{\max} \qquad \forall (i,j) \in A \tag{11}$$

$$\mu \ge \mu_0 \tag{12}$$

$$T_{mn} \le M \qquad \forall (m,n) \in W \tag{13}$$

$$y_s \in \{-1, 1, 2\} \qquad \forall s \in S \tag{14}$$

$$u_s \in \{0, 1\} \qquad \forall s \in S \tag{15}$$

Equation (1) shows the main objective function of the leader problem, which is maximizing the O-D matrix multiplier as the reserve capacity factor. Equations (2)-(4) ensure that direction variables get reasonable values. Inequality (5) restricts the total link expansion cost to the available budget.

For a given demand matrix, each arc flow f_{ij} is a function of the demand multiplier μ, its signal setting variable λ_{ij} and its capacity increment variable u_s. According to inequalities (6)-(7), traffic flow on each arc is restricted by the arc capacity and a flow saturation degree. The arc capacities on each link are defined in (8) and (9), and depend on the flow capacity and the signal setting.

Equation (10) implies that the sum of green time proportions on each signal controlled intersection must be equal to 1. Inequality (11) limits all green time proportions to their given upper and lower bounds. Inequality (12) ensures that the O-D demand matrix multiplier is not lower than a minimum value. Inequality (13) ensures that the minimum travel time between each OD pair is less than a large value M. This implies that there must be at least one path between each OD pair, which means that the network must be strongly connected. If the condition is not met, there is at least one OD pair with very large travel time between them. Finally, constraints (14) and (15) impose settings to variables' domains. The direction variable for each link (y_s) takes one of the following values:

- 1 for an one-way link in a forward direction;
- -1 for an one-way link in a backward direction;
- 2 for a two-way link.

The variable for capacity expansion is a binary variable. For each link $s \in S$, this variable takes the value of 1 if the link is selected for capacity expansion and 0 otherwise.

2.2. Lower level formulation

The lower level optimization model is defined in (16) to (19):

$$Minimize \;\; Z_{II} = \sum_{\substack{(i,j)\in A_s \\ s\in S}} \int_0^{f_{ij}} t_{ij}(x, C_{ij}(\lambda_{ij}, u_s, y_s))dx \tag{16}$$

Subject to:

$$\sum_{r\in R_{mn}} F_r = \mu \; q_{mn} \qquad \forall (m,n)\in W \tag{17}$$

$$f_{ij} = \sum_{r\in R} F_r \; \phi_{ij}^r \qquad \forall (i,j)\in A \tag{18}$$

$$F_r \geq 0, \qquad \forall r\in R \tag{19}$$

The lower level problem is the user equilibrium (UE) assignment problem. For further details, see Sheffi (1985).

3. SOLUTION ALGORITHM

It has been proved that even bi-level problems with only linear constraints are NP-Hard (Hansen et al., 1992). Here, a two-step recursive procedure is applied to solve this bi-level problem. First, for a feasible solution of discrete variables in step I, a bi-level problem is solved during step II. The best solution is returned to step I as a group of known parameters. Then, the first step is solved and its nearly optimum solution is inputted to step II as a group of known parameters and this loop continues till the stopping conditions are met.

3.1. First step of the algorithm

Excluding capacity expansion and signal setting constraints, the upper level formulation is almost similar to Drezner and Wesolowsky (1997) model, which is developed to determine the best configuration of one-way and two-way streets in a network. To solve their problem, Drezner and Wesolowsky (1997) proposed a branch and bound algorithm for small-sized problems and three heuristic algorithms to solve larger problems. Later, Drezner and Salhi (2000) suggested an algorithm based on tabu search (TS) to solve this problem. Drezner and Salhi (2002) compared several heuristic and meta-heuristic methods such as descent algorithm, simulated annealing, genetic algorithm (GA) and TS. They showed that the designed population-based algorithms provide better results compared to other methods. Later, Alvarez et al. (2005) proposed a scatter search approach for solving a general network design problem for undirected networks. They showed that this algorithm is capable for finding good solutions in large-scale problems within a reasonable time. Gallo et al. (2010) used this method for solving a directed network design problem. Moreover, Martí et al. (2005) compared scatter search procedure with GA, in the context of searching for nearly optimal solutions to permutation problems. They observed that the scatter search found solutions with a higher average quality earlier than GA variants. As can be seen from the findings of the previous research, there are successful applications of population-based metaheuristics, and specifically Scatter Search in network design problems. Moreover, this algorithm has been applied in few UNDP researches. Regarding this background, a hybrid scatter search method has been chosen for solving the upper level.

Scatter search is a kind of population based meta-heuristic technique for solving complex optimization problems. This technique includes five steps to update and improve the population by operating on current solution subsets to generate new solution subsets in any iteration.

Step 1. Initial solutions set generation

Generate a set of diverse, feasible and connected solutions from the current network design using the diversification generation method applied in Gallo et al. (2010). A diverse solution is generated by changing the design of predefined number of links, in which the number translates into the distance of the generated solution from the current network. The types of distances and the number of solutions to be generated for each type, are determined beforehand, and then the solutions are generated based according to it. All these solutions (if be connected networks) constitute the initial set which then are used to generate the initial reference set.

Step 2. Reference set generation (and updating)

In this step, the reference set with size P is built or updated. At the first iteration, the initial reference set gets $2/3\times P$ of its members from better solutions in the initial solutions set and $1/3\times P$ of them from the scattered solutions with maximum distances from the best solution (using max-min criterion).

For updating the reference set through the algorithm, the obtained solutions from step 4 (and 5) are combined with the current reference set. Then $2/3\times P$ of the best solutions, and $1/3\times P$ scattered solutions from the combined set are selected to form the new reference set. Then the best solution found so far, is compared with the best solution achieved in the current reference set and is updated if necessary.

In the predefined number of iterations, if the difference between the objective function value of the best solution in the current iteration and the best solution found so far, is less than a predefined small value, the algorithm stops. Otherwise, it proceeds to step 3.

Step 3. Solutions subset generation

The reference set solutions are used to form different subsets. Firstly, all binary subsets are generated. Next, all subsets with three members are formed from the collection of binary subsets and the best reference set solution not contained in them. The same method is used to generate all subsets consisting of *i* members are generated for $i = 4$ to P.

For large networks, since the number of all subsets is huge, the desired number of subsets is fixed beforehand. The algorithm starts building subsets as the above, until their number reaches the predefined value.

Step 4. Solution combination

The members of each subset are used to generate a new solution. If K is the total number of subsets, then K new solutions are generated. Each variable value of a new solution is derived from one of the subset members, using the famous roulette wheel function in GA through which, better solutions have more chance to dedicate their elements to the new solution.

Step 5. Improvement in the current solutions

In this step, the best of the K generated solutions is subjected to random search method in its neighborhood, to get a local optimum solution. The method randomly changes the link capacities of the network (not their direction). Then the algorithm proceeds to step 2 to start the next iteration, i.e. update the reference set.

In steps 4 and 5, a connectivity test is performed for any new solution, and a disconnected solution is discarded and replaced with another one. A shortest path generation method, Dijkstra's algorithm, is used to check the connectivity of all O-D pairs in each network.

3.2. Second step of the algorithm

As mentioned before, by fixing the nearly optimal values of discrete variables obtained in the first step, the resultant problem becomes a known bi-level problem with a simple linear objective function, some linear signal setting constraints, some nonlinear implicit capacity constraints in the upper level problem and the user equilibrium problem in the lower level problem.

A line search algorithm is embedded in the proposed scatter search procedure to cope with capacity constraints. For a given UE flow obtained from the lower level problem, signal timing optimization problem is to be solved with Green Time Swapping Algorithm (GTSA) which is a local search algorithm to get signal setting optimal vector for any UE flow. This procedure searches for a balance condition between green phases of different stages in a signalized intersection in order to swap green time from less pressurized stages to more pressurized stages until the related phases have the same pressure level. The algorithm has been described completely in Lee and Machemehl (2005).

After signal setting characterization for UE flows in the previous section, the network reserve capacity could be updated using a golden section search (GSS) method. Before applying this procedure, we need to prove that the optimal solutions exist in the boundary of capacity constraints.

In the following, some lemmas are proved to show the existence of optimal solutions on the boundary of implicit capacity constraints (6) and (7). In the absence of signal setting variables, these constraints can, respectively, be changed to (20) and (21):

$$f_{ij}(\mu,u_s) \le p_{ij} C_{ij}(u_s, y_s) \qquad \forall s \in S, (i,j) \in A_s \tag{20}$$

$$f_{ji}(\mu,u_s) \le p_{ji} C_{ji}(u_s, y_s) \qquad \forall s \in S, (i,j) \in A_s \tag{21}$$

In these constraints, y_s and u_s have been taken from the upper level problem as constant values. According to Yang and Bell (1998a), the optimal value of system reserve capacity is located at the boundary of (20) and (21). In our situation of having signal setting variables within capacity constraints as used in our model (constraints (6) and (7)), we need to find how these signal setting variables affect the optimal value of the reserve capacity. Thus, we will get it by proving two lemmas as follows.

Lemma 1. The relationship between the arc flow (f_{ij}) and the arc signal setting variable (λ_{ij}) is as follows:

$$\lambda_{ij} = \frac{a' \times \sqrt[d']{f_{ij}^{\,d'} + b'}}{\sqrt[d']{f_{ij}^{\,d'} + b'} + c'} \tag{22}$$

where a', b', c' and d' are nonnegative constants.

Lemma 2. The derivative $\dfrac{d\lambda_{ij}}{df_{ij}}$ is equal or less than 1. It means:

$$d\lambda_{ij} \le df_{ij} \quad \text{or} \quad \frac{d\lambda_{ij}}{df_{ij}} \le 1 \tag{23}$$

The proofs of these lemmas are provided in the Appendix.

By these proofs, we find the relationship between the signal setting and flow variables. According to Lemma 2, when increasing the reserve capacity of the network, it suffices to consider its effect only on arc flows. The effect on signal setting variables could be ignored, since it is in line with the effect on flows according to Lemma 1, but has a much smaller impact than them according to Lemma 2. In other words, the reserve capacity of system is only the function of arc flows obtained by solving the UE problem in the lower level. Therefore, the optimal value of system reserve capacity is located at the boundary of (6) and (7). To obtain this optimal value, a line search algorithm called GSS is used as follows:

Step 1. Consider an initial value for the reserve capacity $\mu = \mu_0$ and $n = 0$.

Step 2. Obtain the flow of each arc $f_{ij}(\mu_n)$ by solving an UE problem in the lower level according to the reserve capacity of system.

Step 3. Calculate the difference between arc flow and capacity for each arc (i, j),

$\tilde{a}_{ij} = f_{ij}(\mu_n) - C_{ij}$.

Step 4. If all $\tilde{a}_{ij}$ s be equal or less than 0, set $\mu_{n+1} = \mu_n (1 + \rho)$, where ρ is a small positive value, and $n = n + 1$, then go to step 2, else set $\mu_{left} = \mu_{n-1}$, $\mu_{right} = \mu_n$ and calculate

$\mu_{mean} = \dfrac{\mu_{left} + \mu_{right}}{2}$ and set $\mu_{n+1} = \mu_{mean}$.

Step 5. If $|\mu_{n+1} - \mu_n| \leq 0.001$, set $\mu^* = \mu_{mean}$ and stop the algorithm, else obtain arc flows by solving an UE problem in the lower level according to the reserve capacity of system $f_{ij}(\mu_{mean})$.

Step 6. Calculate the difference between the flow and capacity for each arc (i, j),

$\tilde{a}_{ij} = f_{ij}(\mu_{mean}) - C_{ij}$.

Step 7. If all $\tilde{a}_{ij}$ s be equal or less than 0, set $\mu_{left} = \mu_{mean}$ and $\mu_{mean} = \dfrac{\mu_{left} + \mu_{right}}{2}$, then go to step 5, else set $\mu_{left} = \mu_{mean}$, $\mu_{right} = \mu_n$ and calculate $\mu_{mean} = \dfrac{\mu_{left} + \mu_{right}}{2}$, then go to step 5.

3.3. Comparison with Genetic Algorithm

For evaluating the performance of our proposed HSS algorithm, its performance was compared with a standard genetic algorithm (SGA) with original version specifications. Like scatter search, GA is a population-based metaheuristic which was first introduced by Holland (1975). GA has been successfully applied in the network design problems (e.g., Yin, 2000; Drezner and Salhi, 2002; Drezner and Wesolowsky, 2003; Cantarella et al., 2006; Chen et al., 2006).

3.4. The main attributes of SGA

The genetic algorithm used in this paper, is the simple version of GA, with a specific crossover operator adopted from the literature. The main attributes of the algorithm are as follows:

- The initial population set generation method is similar to step 1 of HSS.
- At each generation, 2 parents are selected using the roulette wheel function.
- The selected parents are subjected to crossover operator similar, to the one in developed in Drezner and Wesolowsky (1997). This operator uses each of the N network nodes to generate two selection patterns for deciding to select which link design from which parent. Thus, the operator can produce at most $2 \times N$ feasible offspring if all have connected networks.
- There are two kinds of mutation operators in this algorithm; firstly, a predefined number of

randomly selected offspring are subjected to mutation only on their link directions and secondly, a mutation operator randomly perturbs the link capacities of the best solution in the population, after it is updated.

- The population evolution is carried out by substituting the offspring set with the same number of worst solutions.
- The stopping criterion is the same as HSS.

Table 3 provides an overall view to the proposed HSS and SGA and their general features. The details will be described later.

Table 3. Comparison of developed algorithms.

Algorithm	Number of Iterations	Solution Generation Method	Evolution Strategy	Stopping Criteria
HSS	*T* iterations	Generating a reference set using diversification generation method, then generating and combining and improving subsets	Selecting a set of best solutions and most scattered solutions from the combination of the current reference set and the new generated and improved solutions, and building the new reference set using from the combined set	The best solution of the current iteration minus the best known solution so far < ε (for a predefined number of iterations)
SGA	*G* generations	Selecting two parents, applying crossover and mutate some superior solutions	Replacing a number of worst population solutions with offspring solutions	The best solution of the current iteration minus the best known solution so far < ε (for a predefined number of iterations)

4. NUMERICAL RESULTS

In this section, numerical results have been obtained for a small test network and two real networks. At the first step, the developed model and the algorithm were used for a small test network to verify the procedure accuracy comparing results with an exact solution algorithm outputs. After this verification test, the procedure has been applied to two real networks to illustrate the algorithm performance and applicability of the proposed solution method to realistic applications. We could not prove algorithm efficiency by comparing our solution algorithm with other previous ones, because the MNDP with this configuration has not been proposed in previous researches. Therefore, we used a standard GA, as one of the traditional and commonly used algorithm for benchmarking, to solve this problem for small and medium sized networks. For both examples, the algorithm was run many times individually to obtain results. In fact, the algorithm was run 30 times for small-sized and 15 times for medium-sized networks. All computational processes have been done on a laptop with an Intel(R) Core(TM) 2 Dou CPU and a 3GB RAM configuration.

4.1. Parameter setting

The parameters of the two algorithms were set by using series of experiments, by searching for parameter ranges in similar algorithms from related papers to find some initial ranges. The parameters of the two algorithms were set so that their computational efforts for solving each test network are as close as possible to each other, for the purpose of fair comparison. This is done for both small and medium sized example networks. Table 4 depicts the parameter settings of the algorithms for each test network.

Table 4. Parameter setting for the algorithms.

Algorithm	Parameters	Small Network	Sioux Falls City Network	Friedrichshain Center Network
HSS	*P* (size of the reference set)	9	36	130
	K (total number of generated solutions)	9	36	400
	Max number of iterations for (the best solution in the current iteration – the best known solution so far) < ε	20	50	100
	ε (Epsilon)	1.000e-04	1.000e-02	1.000e-01
SGA	*P* (size of the population set)	9	25	
	Number of mutated offspring	3	12	
	Max number of iterations for (the best solution in the current iteration – the best known solution so far) < ε	20	80	-
	ε (Epsilon)	1.000e-04	1.000e-02	

4.2. Results for simple network

One small network is shown in Figure 2. This network was introduced by Wong and Yang (1997) and has two O-D pairs, 7 links and 6 nodes, of which nodes 2 and 5 are signal-controlled intersections. The current O-D demand from nodes 1 to 6 is 18 veh/min, from nodes 3 to 4 is 6 veh/min. This network example has been used in the literature. In this network, two nodes 2 and 5 are signal controlled junctions, so relations between signal control variables are as below:

$\lambda_{1,2} = \lambda_{5,2} = 1 - \lambda_{3,2} = 1 - \lambda_{4,2}$ and $\lambda_{2,5} = \lambda_{6,5} = 1 - \lambda_{3,5} = 1 - \lambda_{4,5}$ (ignoring lost time for simplicity)

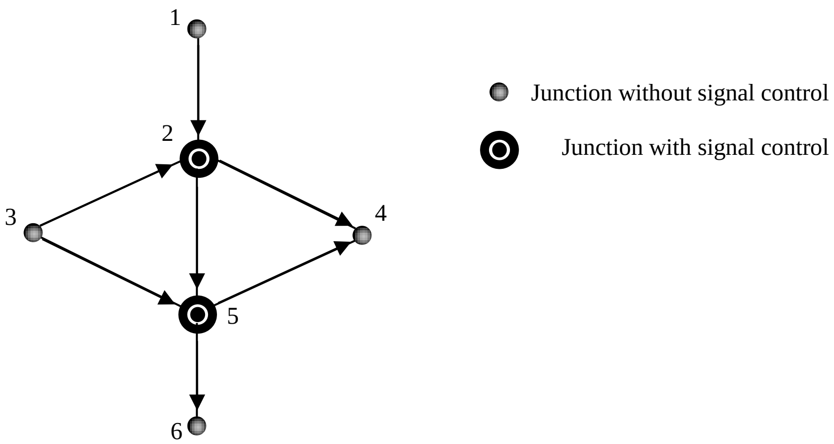

Figure 2. Small example network.

The input parameters for this simple network are shown in Table 5. Note that for all example networks, links are represented by their corresponding nodes, such that the node with smaller number comes first.

Table 5. Parameters of links in the small network.

Link	**1-2**	**2-3**	**2-4**	**2-5**	**3-5**	**4-5**	**5-6**
t_{ij}^0	2	2	1	3	1	2	1
t_{ji}^0	2	2	1	3	1	2	1
c_{ij}	30	24	24	35	30	30	30
c_{ji}	30	24	24	35	30	30	30
d_s	1	1	1	1	1	1	1
μ_0	=1		θ	=366			
η	=100						

To evaluate the capability of the HSS algorithm in achieving optimal solutions, a branch and bound (B&B) method was applied to solve the problem. Since the run time of B&B is very high even for small networks, we have used it only for the small case. For this network, comparative outputs of solving this problem are proposed by the developed HSS, the SGA and B&B method. All three algorithms have reached the optimum solution. The exact algorithm has to search 32670 distinct network designs to find the optimum solution. The two metaheuristics were run 30 times. As a consequence, the best and average results have been provided in Table 6. Note that the reported network designs for this case and the Sioux Falls case, are the improved versions of the initial networks. Although, in the best case the SGA performs slightly better, based on the average results the HSS algorithm is the dominant. Table 6 depicts that all algorithms have reached the optimum solution, although HSS could reach it by searching fewer solutions which indicates it is the preferred solution procedure.

Table 6. The obtained results for solving the simple network.

Link	y_s	u_s	λ_{ij}	λ_{ji}
1-2	1	1	0.541	-
2-3	2	0	-	0.458
2-4	2	0	-	0.458
2-5	-1	0	0.05	0.541
3-5	2	0	0.95	-
4-5	2	1	0.95	-
5-6	1	0	-	0.05

Solving algorithm	**HSS**		**SGA**		**B&B**
	Best	**Average**	**Best**	**Average**	**-**
Matrix multiplier	2.18	2.18	2.18	2.18	2.18
Total number of iterations	1	5.4	1	5.9	32670
Total computation runtime (s)	6	15.3	5	21	6754
Total number of generated solutions	53	73	22	128	-

4.3. Results for Sioux Falls city network

In this section, Sioux Falls city network is used as the first real example to test our developed solution algorithm. Sioux Falls city network shown in Figure 3 is a signalized network defined by a graph with 24 O-D pairs, 24 nodes and 38 links. The initial values of the parameters such as the free flow travel time, capacity, and length of each links have been taken from the website provided by Bar-Gera (2013). The following values were used for the required parameters: $\mu_0 = 1$, $\eta = 30000$, $\theta = 336$.

As mentioned before, to confirm the correctness of the results, the problem has been solved by the proposed HSS and the SGA. Both solving procedures were run 15 times. The results are shown in Table 7 which indicates that the best obtained matrix multipliers in both algorithms are equal. Although the average run time of HSS for each iteration does not exceed its counterpart in SGA, it reached those results in significantly less iterations.

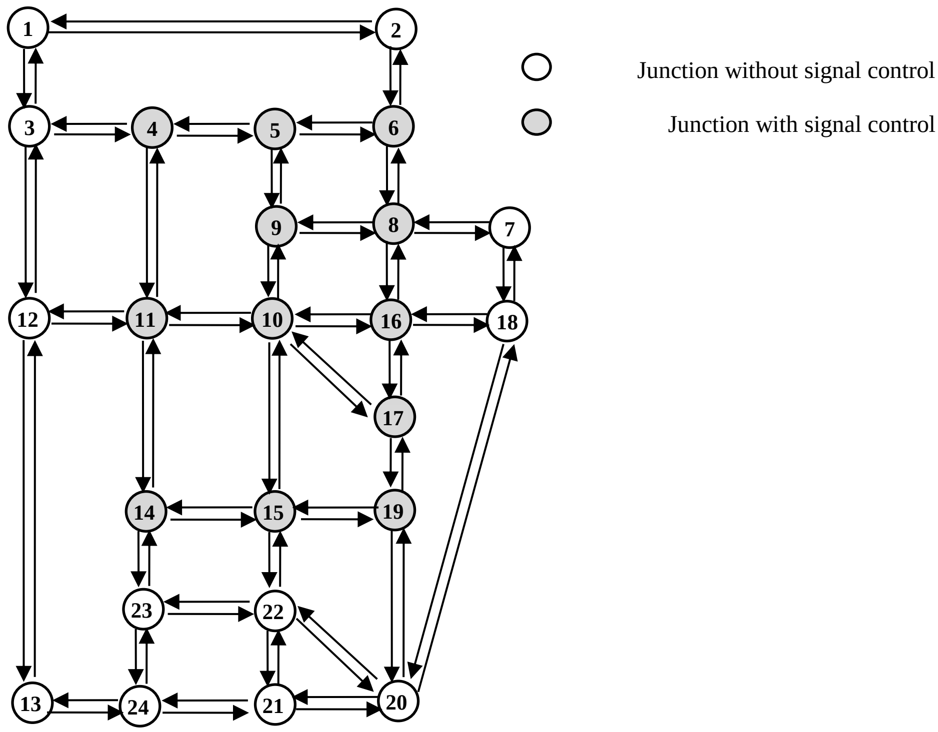

Figure 3. Sioux Falls network.

The HSS algorithm reached its best result for 53% of times below 100 iterations while the SGA found this result in above 200 iterations in all 15 runs. Consequently, the average total computational time in the HSS procedure is remarkably less than the SGA. Besides, the GA reached its best result only in 40% of its total 15 runs. The results in Table 7 clearly show the superiority of the HSS algorithm comparing the best, the worst and the average results of both algorithms.

Table 7. Comparison between HSS and SGA for Sioux Falls network.

Algorithm	**Results**	**Total Computation Time (s)**	**Total Number of Iterations**	**Avg. Computation Time per Iteration**	**Objective Function Value**
HSS	Best	2664	61	37	1.1621
	Worst	11220	317	35	1.1621
	Average	3844	97	38	1.1621
SGA	Best	9982	217	46	1.1621
	Worst	17653	409	43	1.1270
	Average	13717	292	43	1.1498

We can see the best obtained results among the 15 runs of HSS in Table 8. The total computational procedure for the given results took 44 minutes. In this developed network, nearly 65% of streets have changed to one-way streets. These results show the improvement of the urban network on adopting the solution obtained by the HSS method.

Table 8. The best obtained result for Sioux Falls network.

Links	1-2	1-3	2-6	3-4	3-12	4-5	4-11	5-6	5-9	6-8	7-8	7-18	8-9	8-16	9-10	10-11	10-15	10-16	10-17	11-12	11-14	12-13	13-24	14-15	14-23	15-19	15-22	16-17	16-18	17-19	18-20	19-20	20-21	20-22	21-22	21-24	22-23	23-24
y_s	2	2	1	2	2	-1	1	1	-1	1	2	-1	2	2	-1	-1	-1	-1	-1	-1	1	-1	2	1	1	-1	2	2	1	-1	2	-1	1	2	-1	-1	-1	2
u_s	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
λ_{ij}	-	-	0.6	0.39	-	0	0.49	0.4	0	0.51	0.49	0	0.45	0.41	0.39	0	0	0	0	0	1	-	-	0.6	-	0	0	0.41	-	0	-	0	-	-	-	-	-	-
λ_{ji}	-	-	-	-	-	0.54	0	0	1	0	-	-	0.49	0.51	0	0.36	0.36	0.39	0.25	0.51	0	-	-	0	0	0.6	0.4	0.41	0	0.59	-	1	-	-	-	-	-	-

Objective Function value (Matrix Multiplier)	1.1621
Total Number of Iteration	61
Total computation time (s)	2664
Average iteration time (s)	37

4.4. Results for the Friedrichshain center network

The second real network used in this research is the Friedrichshain center network as a part of Berlin network in Germany. This network is a graph with 266 nodes and 224 links. Initial parameter values such as those for the free flow travel time, capacity, and length of each link are obtained from the website provided by Bar-Gera (2013). In the primary solution, all links are considered as two-way streets. The primary schematic network is shown in Figure 4.

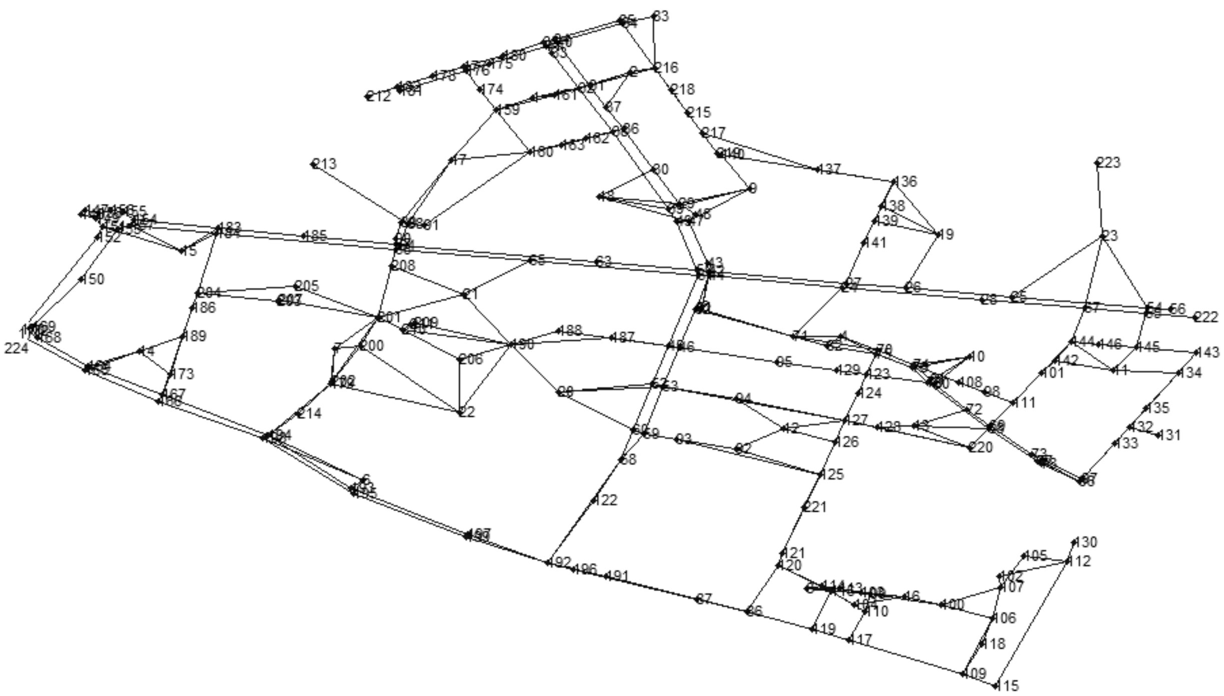

Figure 4. Friedrichshain center network (Asudegi, 2009).

Table 9 shows results of the HSS algorithm. Although all streets in the primary network are to be assumed two-way links, in the final solution about 34% of streets have been changed to one-way.

Table 9 shows the best solution objective function obtained in this procedure. The total computational procedure time for the given results takes about 59 hours.

Table 9. The Best acquired solution result for Friedrichshain network.

Objective function value: matrix multiplier	1.1252
Total number of iterations	211
Number of one-way streets	76 links of total 224 links (about 34%)
Total computation time	212688 seconds (59 hours)
Average iteration time	1008 seconds

The convergence pattern of the proposed HSS in solving the problem for this network is shown in Figure 5.

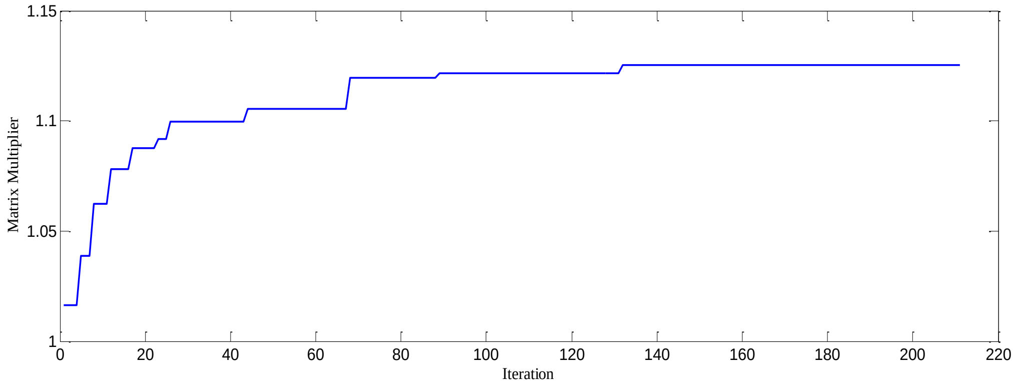

Figure 5. Convergence pattern of HSS algorithm.

5. CONCLUSION AND FUTURE RESEARCH

In this paper, a new bi-level model is developed for a MNDP problem. In the upper level problem, two discrete variables (redesign direction, capacity expansion of links) and one continuous variable (signal setting) are optimized. In the lower level problem, the UE problem is solved. For solving this problem, one hybrid scatter search method incorporating golden section search has been proposed. The algorithm has been tested for one small and two real case studies. To evaluate the capability of the algorithm in achieving the optimum solution in the small network, the result has been compared with the results obtained from the branch and bound method. Moreover, the performance of the proposed hybrid scatter search was illustrated by comparing it with a kind of standard GA for the small and one of the real networks. The results show that, in both cases, the developed HSS has been the absolute dominant algorithm.

Because of the diversity of network planning factors involved in our problem, several assumptions are used in the model, such as a common matrix multiplier for the whole network, simple objective function for the upper level of the model and using simple deterministic UE in the lower level problem. As a suggestion for future research, this proposed framework could be extended by considering a stochastic UE problem in the lower level problem. Besides, a multi-objective function and different matrix multipliers for each O-D pair can be used in the upper level of the model. Finally, one can consider other network decisions such as lane allocation in two-way streets or turning restriction design at intersections for the upper level problem. These extensions will make the problem more complex, and may require designing more efficient algorithms. Strategies such as parallelizing or distributed computing may be able to handle such complexities.

Acknowledgements

This research is partially supported by a grant from the National Natural Science Foundation of China (No. 71271183) and a grant (No. 201411159063) from the University Research Committee of the University of Hong Kong. The authors are grateful to the reviewer for his/her constructive comments.

Appendix

Lemma 1. The relationship between the arc flow (f_{ij}) and the arc signal setting variable (λ_{ij}) is as follows (in what follows, the arc notation (i,j) is shown as (a) for simplicity):

$$\lambda_a = \frac{a' \times \sqrt[d']{f_a^{\,d'} + b'}}{\sqrt[d']{f_a^{\,d'} + b'} + c'} \tag{24}$$

where a', b', c' and d' are nonnegative constants.

Proof: To prove this equation, we use the pressure concept described in Lee and Machemehl (2005). The pressure formularization is different depending on what policy is chosen. Here, we have chosen the form of pressure based on BPR travel time minimization policy. In this regard, the pressure is formulated through (25) where, P_a indicates the pressure of arc a, $t^a{}_0$ and c are the free flow travel time and the saturation flow, respectively, and α and β are constant parameters.

$$P_a = \frac{t_0^a.\alpha.\beta.f_a^{\,\beta+1}}{c^{\beta}.\lambda_a^{\,\beta+1}} \tag{25}$$

Each signalized intersection includes several stages (green periods), as shown in equation (26), the total pressure of stage k in intersection l ($P_{stage\ k}$) is determined by the summation of related arcs' pressures that receives green (P_a), where L_l is the set of all stages in intersection l

$$P_{stage\ k} = \sum_{\substack{arc\ a\ receives\ green \\ at\ stage\ k}} P_a \qquad k \in L_l \tag{26}$$

According to Lee and Machemehl (2005), in the optimal state, the pressures of all stages in one intersection are equivalent. Take one intersection with two stages and two arcs in each stage for example in which arcs 1 and 2 receive green simultaneously, and arcs 3 and 4 receive green simultaneously; the intersection pressure balancing equation is as (27). It means that the total pressure of arcs 1 and 2 is equal to the total pressure of arcs 3 and 4.

$$P_{stage1} = P_{stage2} \Rightarrow P_1 + P_2 = P_3 + P_4 = \frac{t_0^1.\alpha.\beta.f_1^{\beta+1}}{c_1^{\beta}.\lambda_1^{\beta+1}} + \frac{t_0^2.\alpha.\beta.f_2^{\beta+1}}{c_2^{\beta}.\lambda_2^{\beta+1}} = \frac{t_0^3.\alpha.\beta.f_3^{\beta+1}}{c_3^{\beta}.\lambda_3^{\beta+1}} + \frac{t_0^4.\alpha.\beta.f_4^{\beta+1}}{c_4^{\beta}.\lambda_4^{\beta+1}} \tag{27}$$

Equation (28) shows relation between the green time proportions of stages for an intersection with two stages and two arcs belonging to each stage.

$$\lambda_1 = \lambda_2 = 1-\lambda_3 = 1-\lambda_4 \Rightarrow \lambda_{stage1} = 1-\lambda_{stage2} \tag{28}$$

Equation (29) is the consequence of synthesizing (27) and (28) where f_1 to f_4 are UE flows taken from the user equilibrium problem.

$$\frac{t_0^1.\alpha.\beta.f_1^{\beta+1}}{c_1^{\beta}.\lambda_{stage1}^{\beta+1}} + \frac{t_0^2.\alpha.\beta.f_2^{\beta+1}}{c_2^{\beta}.\lambda_{stage1}^{\beta+1}} = \frac{t_0^3.\alpha.\beta.f_3^{\beta+1}}{c_3^{\beta}.(1-\lambda_{stage1})^{\beta+1}} + \frac{t_0^4.\alpha.\beta.f_4^{\beta+1}}{c_4^{\beta}.(1-\lambda_{stage1})^{\beta+1}} \Rightarrow$$

$$\lambda_{stage1} = \lambda_1 = \lambda_2 = \frac{\sqrt[\beta+1]{\left(\frac{t_0^1}{c_1^{\beta}}\right)f_1^{\beta+1} + \left(\frac{t_0^2}{c_2^{\beta}}\right)f_2^{\beta+1}}}{\sqrt[\beta+1]{\left(\frac{t_0^1}{c_1^{\beta}}\right)f_1^{\beta+1} + \left(\frac{t_0^2}{c_2^{\beta}}\right)f_2^{\beta+1}} + \sqrt[\beta+1]{\left(\frac{t_0^3}{c_3^{\beta}}\right)f_3^{\beta+1} + \left(\frac{t_0^4}{c_4^{\beta}}\right)f_4^{\beta+1}}} \tag{29}$$

For intersections with more than two stages and more than two arcs in each stage, the same procedure is repeated. Equation (30) demonstrates the pressure balancing in an intersection with n different stages where the pressure for stage n is shown by $P_{stage\ n}$ and the pressure of arc a belonging to this stage is shown by P_a^n.

$$P_{stage\ 1} = L = P_{stage\ k} = L = P_{stage\ n} \Rightarrow \sum_{i=1}^{m} P_i^1 = L = \sum_{i'=1}^{m'} P_{i'}^k = L = \sum_{i''=1}^{m''} P_{i''}^n \tag{30}$$

where L is the total pressure; m, m' and m'' are, respectively, the number of arcs belonging to the concerned stages.

By rearranging the green time proportion constraint in the intersection, equation (31) shows the relation between the green time proportions of all stages in one intersection.

$$\lambda_{stage\ k} = (1 - \sum_{i \neq stage\ k, i \neq stage\ n} \lambda_i) - \lambda_{stage\ n} \tag{31}$$

Substituting equation (31) in the pressure balance equation (30) will result (32):

$$\lambda_{stage k} = \lambda_a = \frac{a' \times \sqrt[\beta+1]{\left(\frac{t_0^a}{c_a^\beta}\right) f_a^{\ \beta+1} + b'}}{\sqrt[\beta+1]{\left(\frac{t_0^a}{c_a^\beta}\right) f_a^{\ \beta+1} + b'} + c'} \tag{32}$$

In (32), $\lambda_{stage\ k}$ is the green time proportion of stage k which is equal to signal setting variables belonged to this stage such as λ_a. Besides, a', b', and c' obviously have nonnegative values. In these relations, L_l is set of all stages within intersection l and S_k indicates set of all arcs belonged to stage k.

$$a' = (1 - \sum_{i \in L_l \setminus k} \lambda_i) \tag{33}$$

$$b' = \sum_{j \in S_k \setminus a} \left(\frac{t_0^j}{c_j^\beta}\right) f_j^{\ \beta+1} \tag{34}$$

$$c' = \sum_{i \in L_l \setminus k} \sqrt[\beta+1]{\sum_{j \in S_i} \left(\frac{t_0^j}{c_2^j}\right) f_j^{\ \beta+1}} \tag{35}$$

Lemma 2. The derivative $\frac{d\lambda_a}{df_a}$ is equal or less than 1. It means:

$$d\lambda_a \le df_a \quad \text{or} \quad \frac{d\lambda_a}{df_a} \le 1 \tag{36}$$

Proof: We use the result of lemma 1 for the sample intersection to prove the second lemma in (37).

$$\frac{d\lambda_1}{df_1} = \frac{d\left(\frac{\sqrt[\beta+1]{f_1^{\ \beta+1} + f_2^{\ \beta+1}}}{\sqrt[\beta+1]{f_1^{\ \beta+1} + f_2^{\ \beta+1}} + \sqrt[\beta+1]{f_3^{\ \beta+1} + f_4^{\ \beta+1}}}\right)}{df_1} \tag{37}$$

$$\frac{d\lambda_1}{df_1}=\frac{\dfrac{(\beta+1)\,f_1^{\,\beta}\,(\sqrt[\beta+1]{f_1^{\,\beta+1}+f_2^{\,\beta+1}}+\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}})}{(\beta+1)\sqrt[\beta+1]{\left(f_1^{\,\beta+1}+f_2^{\,\beta+1}\right)^{\beta}}}-\dfrac{(\beta+1)\,f_1^{\,\beta}\,(\sqrt[\beta+1]{f_1^{\,\beta+1}+f_2^{\,\beta+1}})}{(\beta+1)\sqrt[\beta+1]{\left(f_1^{\,\beta+1}+f_2^{\,\beta+1}\right)^{\beta}}}}{(\sqrt[\beta+1]{f_1^{\,\beta+1}+f_2^{\,\beta+1}}+\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}})^2}$$

$$\frac{d\lambda_1}{df_1}=\frac{\dfrac{f_1^{\,\beta}}{\sqrt[\beta+1]{\left(f_1^{\,\beta+1}+f_2^{\,\beta+1}\right)^{\beta}}}\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}}}{(\sqrt[\beta+1]{f_1^{\,\beta+1}+f_2^{\,\beta+1}}+\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}})^2}$$

$$\frac{d\lambda_1}{df_1}=\frac{(\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}})\;f_1^{\,\beta}}{\sqrt[\beta+1]{\left(f_1^{\,\beta+1}+f_2^{\,\beta+1}\right)^{\beta}}\,(\sqrt[\beta+1]{f_1^{\,\beta+1}+f_2^{\,\beta+1}}+\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}})^2}$$

$$\frac{d\lambda_1}{df_1}=\frac{b''f_1^{\,\beta}}{a''(c''+b'')^2}$$

where

$$\begin{cases} a''=\sqrt[\beta+1]{\left(f_1^{\,\beta+1}+f_2^{\,\beta+1}\right)^{\beta}}\;\geq\;f_1^{\,\beta}\;\geq 0 \\[2ex] b''=\sqrt[\beta+1]{f_3^{\,\beta+1}+f_4^{\,\beta+1}}\geq 0 \\[2ex] c''=\sqrt[\beta+1]{f_1^{\,\beta+1}+f_2^{\,\beta+1}}\;\geq\;f_1^{\,\beta}\;\geq 0 \\[2ex] f_1\geq 0 \end{cases}\quad\Rightarrow\quad 0\leq\frac{b''f_1^{\,\beta}}{a''(c''+b'')^2}\leq 1$$

Assuming an intersection with more than two stages and more than two arcs entering the intersection in each stage, the derivative of $\lambda_k\,/\,f_a^{\,k}$ is as follows, where λ_k is the signal setting variable of stage k and f_a^k is the flow of arc a belonging to stage k.

$$
\begin{cases}
\dfrac{d\lambda_k}{df_a^k} = \dfrac{a''' \times c''' \times f_a^{k\,\beta}}{(\sqrt[\beta+1]{f_a^{k\,\beta+1} + b'''} + c''') \times \sqrt[\beta+1]{(f_a^{k\,\beta+1} + b''')^{\beta}}} \\
0 \le a''' = (1 - \sum_{i \ne k} \lambda_i) \le 1 \\
b''' = \sum_{j \in L_k \ne a} f_j^{k\,\beta+1} \ge 0 \\
c''' = \sum_{i \ne k} \sqrt[\beta+1]{\sum_{j \in L_i} f_j^{i\,\beta+1}} \ge 0 \\
f_a^k \ge 0
\end{cases}
\Rightarrow \quad 0 \le \frac{d\lambda_k}{df_a^k} \le 1
$$

This completes the proof.

REFERENCES

Allsop, R.E. (1972). Estimating the Traffic Capacity of a Signalized Road Junction. *Transportation Research*, *6*(3), 245-255.

Alvarez, A.M., González-Velarde, J.L., De-Alba, K. (2005). Scatter Search for Network Design Problem. *Annals of Operations Research*, *138*, 159-178.

Asudegi M. (2009). Optimal Number and Location of Bluetooth Sensors for Travel Time Data Collection in Networks. *Master Thesis*, *Department of Civil Engineering*, *University of Maryland*, *Maryland*, *United States*,

Bar-Gera, H. (2013). Transportation Network Test Problems, http://www.bgu.ac.il/~bargera/tntp/ [accessed on 30 May 2014]

Cantarella, G.E., Pavone, G., Vitetta, A. (2006). Heuristics for Urban Road Network Design: Lane Layout and Signal Settings. *European Journal of Operational Research*, *175*(3), 1682-1695.

Chen, A., Subprasom, K., Ji, Z. (2006). A Simulation-Based Multi-Objective Genetic Algorithm (SMOGA) Procedure for BOT Network Design Problem. *Optimization and Engineering Journal*, *7*(3), 225-247.

Dimitriou, L., Tsekeris, T., Stathopoulos, A. (2008). Genetic Computation of Road Network Design and Pricing Stackelberg Games with Multi-class Users. In M. Giacobini et al. (Eds.), *Applications of Evolutionary Computing* (pp. 669-678), Springer, Berlin/ Heidelberg.

Drezner, Z., Salhi, S. (2000). Selecting a Good Configuration of One-Way and Two-Way Routes Using Tabu Search. *Control and Cybernetics*, *29*(3), 725-740.

Drezner, Z., Salhi, S. (2002). Using Hybrid Metaheuristics for the One-Way and Two-Way Network Design Problem. *Naval Research Logistics*, *49*, 449-463.

Drezner, Z., Wesolowsky, G.O. (1997). Selecting an Optimum Configuration of One-Way and Two-Way Routes. *Transportation Science*, *31*(4), 386-394.

Drezner, Z., Wesolowsky, G.O. (2003). Network Design: Selection and Design of Links and Facility Location. *Transportation Research Part A*, *37*(3), 241-256.

Farahani, R.Z., Miandoabchi, E., Szeto, W.Y., Rashidi, H. (2013). A Review of Urban Transportation Network Design Problems. *European Journal of Operational Research*, *229*(2), 281–302.

Friesz, T. L. (1985). Transportation Network Equilibrium, Design and Aggregation: Key Developments and Research Opportunities. *Transportation Research Part A*, *19*(5-6), 413-427.

Gallo, M., D'Acierno, L., Montella, B. (2010). A Meta-Heuristic Approach for Solving the Urban Network Design Problem. *European Journal of Operational Research*, *201*(1), 144-157.

Gao, Z., Wu, J., Sun, H. (2005). Solution Algorithm for the Bi-Level Discrete Network Design Problem. *Transportation Research Part B*, *39*(6), 479-495.

Hansen, P., Jaumard B., Savard, G. (1992). New Branch-and-Bound Rules for Linear Bilevel Programming, *SIAM Journal on Scientific and Statistical Computing*, *13*(5), 1194-1217.

Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Michigan.

Jiang, Y., Szeto, W.Y. (2015). Time-Dependent Transport Network Design that Considers Health Cost. *Transportmetrica A*, *11*(1), 74-101.

LeBlanc, L.J. (1975). An Algorithm for the Discrete Network Design Problem. *Transportation Science*, *9*(3), 183-199.

Lee, C., Machemehl, R.B. (2005). *Combined Traffic Signal Control and Traffic Assignment: Algorithms*, *Implementation and Numerical Results*, Center for Transportation Research University of Texas at Austin.

Lee, C.K., Yang, K.I. (1994). Network Design of One-Way Streets with Simulated Annealing. *Papers in Regional Science*, *73*(2), 119-134.

Lo, H.K., Szeto, W.Y. (2009). Time-Dependent Transport Network Design under Cost-Recovery. *Transportation Research Part B*, *43*(1), 142-158.

Long, J.C., Gao, Z.Y., Zhang, H.Z., Szeto, W.Y. (2010). A Turning Restriction Design Problem in Urban Road Networks. *European Journal of Operational Research*, *206*(3), 569-578.

Long, J.C., Szeto, W.Y., Huang, H.J. (2014). A Bi-Objective Turning Restriction Design Problem in Urban Road Networks. *European Journal of Operational Research*, *237*(2), 426-439.

Magnanti, T.L., Wong, R.T. (1984). Network Design and Transportation Planning: Models and Algorithms. *Transportation Science*, 18(1), 1-55.

Martí, R., Laguna, M., Campos, V. (2005). Scatter Search vs. Genetic Algorithms: An Experimental Evaluation with Permutation Problems, In Sharda, R., Voß, S., Rego, C., Alidaee, B. (Eds.), *Metaheuristic Optimization via Memory and Evolution: Tabu Search and Scatter Search* (pp.263-282), Kluwer Academic Publishers, Boston, USA.

Miandoabchi, E., Daneshzand, F., Farahani, R.Z., Szeto, W.Y. (2015). Time-Dependent Discrete Road Network Design with both Tactical and Strategic Decisions. *Journal of the Operational Research Society*, *66*, 894-913.

Miandoabchi, E., Daneshzand, F., Szeto, W.Y., Farahani, R.Z. (2013). Multi-objective Discrete Urban Road Network Design. *Computers and Operations Research*, *40*(10), 2429-2449.

Miandoabchi, E., Farahani, R.Z. (2011). Optimizing Reserve Capacity of Urban Road Networks in a Discrete Network Design Problem. *Advances in Engineering Software*, *42*(12), 1041-1050.

Miandoabchi, E., Farahani, R.Z., Szeto, W.Y., (2012a). Bi-objective Bimodal Urban Road Network Design Using Hybrid Metaheuristics. *Central European Journal of Operations Research*, *20*(4), 583-621.

Miandoabchi, E., Farahani, R.Z., Dullaert, W., Szeto, W.Y. (2012b). Hybrid Evolutionary Metaheuristics for Concurrent Multi-Objective Design of Urban Road and Public Transit Networks. *Networks and Spatial Economics, 12*(3), 441-480.

Migdalas, A. (1995). Bilevel Programming in Traffic Planning: Models, Methods and Challenge. *Journal of Global Optimization, 7*(4), 381-405.

Poorzahedy, H., Abulghasemi, F. (2005). Application of Ant System to Network Design Problem. *Transportation, 32*(3), 251-273.

Poorzahedy, H., Rouhani, O.M. (2007). Hybrid Meta-Heuristic Algorithms for Solving Network Design Problem. *European Journal of Operational Research, 182*(2), 578-596.

Poorzahedy, H., Turnquist, M.A. (1982). Approximate Algorithms for the Discrete Network Design Problem. *Transportation Research Part B, 16*(1), 45-55.

Sheffi, Y. (1985). *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Prentice-Hall, Englewood Cliffs, NJ, USA.

Steenbrink, P.A. (1974). Transport Network Optimization in the Dutch Integral Transportation Study. *Transportation Research*, *8*(1), 11-27.

Sun, H., Gao, Z.Y., Szeto, W.Y., Long, J.C., Zhao, F. (2014). A Distributionally Robust Joint Chance Constrained Optimization Model for the Dynamic Network Design Problem under Demand Uncertainty. *Networks and Spatial Economics*, *14*(3-4), 409-433.

Szeto, W.Y., Lo, H.K., (2006). Transportation Network Improvement and Tolling Strategies: the Issue of Intergeneration Equity. *Transportation Research Part A*, *40*(3), 227-243.

Szeto, W.Y., Lo, H.K., (2008). Time-dependent Transport Network Improvement and Tolling Strategies. *Transportation Research Part A, 42*(2), 376-391.

Szeto, W.Y., Jiang, Y., Wang, D.Z.W., Sumalee, A. (2015). A Sustainable Road Network Design Problem with Land Use Transportation Interaction over Time. *Networks and Spatial Economics*, DOI: 10.1007/s11067-013-9191-9.

Szeto, W.Y., Jaber, X.Q., O'Mahony, M. (2010). Time-dependent Discrete Network Design Frameworks Considering Land Use. *Computer-Aided Civil and Infrastructure Engineering*, 25(6), 411–426.

Szeto, W.Y., Wang, Y., Wong, S.C. (2014). The Chemical Reaction Optimization Approach to Solving the Environmentally Sustainable Network Design Problem. *Computer-Aided Civil and Infrastructure Engineering*, *29*(2), 140-158.

Webster, F.V., Cobbe, B.M. (1966). Traffic signal, *Road Research Technical Paper*, No.*56*, HMSO, London, UK.

Wong, S.C. (1996). On the Reserve Capacities of Priority Junctions and Roundabouts. *Transportation Research Part B*, *30*(6), 441-453.

Wong, S.C., Yang, H. (1997), Reserve Capacity of a Signal-Controlled Road Network, *Transportation Research Part B*, *31*(5), 397-402.

Yang, H., Bell, M.G.H. (1998a). Models and Algorithms for Road Network Design: A Review and Some New Developments. *Transport Review*, *18*(3), 257-278.

Yang, H., Bell, M.G.H. (1998b). A Capacity Paradox in Network Design and How to Avoid It. *Transportation Research Part A*, *32*(7), 539-545.

Yang, H., Wang, J.Y.T. (2002). Travel Time Minimization versus Reserve Capacity Maximization in the Network Design Problem. *Transportation Research Record*, *1783*, 17-26.

Yin, Y. (2000). Genetic-Algorithm-Based Approach for Bi-Level Programming Models. *Journal of*

Transportation Engineering (ASCE), *126*(2), 115-120.

Zhang, H., Gao, Z. (2009). Bilevel Programming Model and Solution Method for Mixed Transportation Network Design Problem. *Journal of Systems Science and Complexity*, *22*, 446-459.

Ziyou, G., Yifan, S. (2002). A Reserve Capacity Model of Optimal Signal Control with User-Equilibrium Route Choice. *Transportation Research Part B, 36*(4), 313-323.