# A novel discrete network design problem formulation and its global optimization solution algorithm

**David Z.W. Wang*, Haoxiang Liu**

School of Civil and Environmental Engineering, Nanyang Technological University, 50 Nanyang Avenue, 639798, Singapore

**W. Y. Szeto**

Department of Civil Engineering, The University of Hong Kong, Pok Fu Lam, Hong Kong

## ABSTRACT

Conventional discrete transportation network design problem deals with the optimal decision on new link addition, assuming the capacity of each candidate link addition is predetermined and fixed. In this paper, we address a novel yet general discrete network design problem formulation that aims to determine the optimal new link addition and their optimal capacities simultaneously, which answers the questions on whether a new link should be added or not, and if added, what should be the optimal link capacity. A global optimization method employing linearization, outer approximation and range reduction techniques is developed to solve the formulated model.

Key words: Network design problem, User equilibrium, Mixed-integer linear programming, Global optimization, Range reduction

## 1. Introduction

The discrete network design problem (DNDP) involves the optimal decision on addition of new links or roadway segments to an existing transportation network, subject to a limited investment budget. Traditionally, given a group of candidate links with fixed capacities, the DNDP is formulated as 0-1 decision problem aiming to determine the optimal road construction plan. The objective of DNDP is to optimize transportation network performance while considering the drivers' routing behavior, for example, following deterministic user equilibrium (DUE) (Sheffi, 1985). The DNDP is typically formulated as a bi-level program with the upper-level minimizing the total travel time cost and the lower-level describing the equilibrium flow pattern.

The DNDP has been widely investigated in previous research works, and it is widely recognized as one of the most difficult frontiers in transportation study due to its computational difficulties in solving the mixed-integer nonlinear nonconvex, bi-level program formulation. Yang and Bell (1998) reviewed a number of models and solution algorithms for network design problem (NDP) based on bi-level programming. Magnanti and Wong (1984) presented a unifying framework for deriving a bunch of algorithms for DNDP and reviewed some computational experience in solving NDP. LeBlanc (1975) proposed a branch-and-bound (B&B) algorithm for solving the upper-level problem of DNDP. Poorzahedy and Turnguist (1982) adopted a well-behaved function to substitute the original total user cost objective function and formulated a single-level model. A B&B based heuristic algorithm was also given in their research. By applying the concept of support function to express the relationship between improvement flows and new addition links, Gao et al. (2005) transformed the bi-level programming of DNDP into a general nonlinear problem and thus traditional constrained optimization algorithms can be used. Solanki et al. (1998) decomposed the DNDP into a sequence of sub-problems and presented a quasi-optimization heuristic algorithm. Furthermore, heuristic/meta-heuristic approaches were studied to solve DNDP, including ant system/cooperating agents algorithm (Poorzahedy and Abulghasemi, 2005), genetic algorithms (Drezner and Wesolowsky, 2003; Kim and Kim, 2006) and so on. Some methods of hybrid meta-heuristic were also designed and compared among each other (Poorzahedy and Rouhani, 2007). More recently, global optimal algorithms for NDP have generated interest amongst researchers. Wang and Lo (2010) employed single-level mixed-integer linear programming (MILP) to approximating continuous network design problem (CNDP), which dealt with continuous expansion of existing links. The nonlinearity of travel time function was removed by applying a convex-combination based piecewise linear approximation. Luathep et al. (2011) further extended this method to solve mixed network design problem (MNDP), which is a combination of CNDP and DNDP. The DUE condition was depicted by a variational inequality (VI) problem and a cutting constraint based algorithm was proposed to seek the optimal solution. Farvaresh and Sepehri (2011) developed a single-level mixed-integer linear programming by transforming the lower-level DUE constraints into the equivalent Karush-Kuhn-Tucker (KKT) condition. Li et al. (2012) presented a global optimal approach for CNDP based on the concept of gap function and penalty. Wang et al. (2013) developed a NDP model with discrete multiple capacity levels to address the problem of adding an optimal number of lanes to existing candidate links. Furthermore, Fontaine and Minner (2014) proposed a solution method based on bender decomposition to solve linearized discrete network design problem. A global optimal method is designed by making use of the relationship between user equilibrium traffic assignment and system optimal principle. Szeto et al. (2014) address a sustainable road network design problem with land use transportation interaction over time. Liu and Wang (2015) proposed a global optimization solution approach for CNDP with stochastic user equilibrium travel flow pattern.

In previous studies, the discrete network design problems (DNDP) assume pre-determined road capacity for candidate link addition, while only addressing the issue that whether or not a new link will be constructed. However, it is more interesting to answer the question that whether or not a new link should be added, and simultaneously, if added, what is the optimal link capacity. In this paper, we exploit a DNDP problem with consideration of link capacity optimization, which aims to optimize the network performance via determining which links should be added from a set of candidate links and what capacities the new links to be constructed should have. The decision variables for a candidate link simultaneously include both discrete (binary) variables, which indicates whether the candidate link will be added or not, and continuous variables, i.e. the link capacity variables (the scenario with only discrete capacity levels is also considered in this paper). The DUE condition is used to describe the equilibrium traffic flow. Taking the advantage of variational inequality formulation in representing the DUE condition, this study firstly formulates a mathematical program with equilibrium constraints. Then, a global optimization method is proposed to solve the problem. As the transport network design problem is naturally formulated as an inherently nonlinear and non-convex problem, the advantage and benefit of finding the globally optimal solution is obvious, to ensure that the network design plan is exactly the “best plan” to achieve the targeted goal. Indeed, no previous studies have ever developed global optimization solution method for solving the transport network design problem presented in this paper, and this study could contribute in filling in this research gap in the literature. Noting that the nonlinearity of the problem stems from the bilinear terms and nonlinear travel time functions in the programming, this study applies two different techniques to deal with them. For the bilinear functions, we apply a Reformulation-linearization technique (Sherali and Adams, 1994, 1998) to transform them into a set of equivalent linear constraints; meanwhile, for the multi-variable travel time functions, we firstly take logarithm of them and then derive its mixed-integer linear relaxation through an outer-approximation technique. By doing so, a mixed-integer linear program (MILP) relaxation model is obtained, whose solution provides a tight lower bound of the original model solution. Then, a range reduction technique is applied to update and improve the lower bound until the gap between the lower bound and upper bound fulfills certain stopping criteria. The solution algorithm is proved to converge to the global optimal solution of the original problem.

This study considers a novel, yet more general NDP problem, which is sought to provide transportation network planners more indicative information not only on new candidate link additions, but also on optimal capacity of the new links, which are otherwise assumed to be given in previous DNDP studies. The developed model is more general formulation, which may include other conventional network design problems as particular cases. For example, when the capacity for each new link addition is given, this model will reduce to traditional DNDP in the literature; when the discrete variables on new link addition plan is predetermined, this problem is indeed a classical continuous network design problem (CNDP). Assuming road capacities to be continuous, the solutions of CNDP provide a “first-best” road capacity expansion plan. In practice, the CNDP modeling and solution algorithm is more useful when signalization or ramp metering is considered (Yang and Bell, 1998). Besides, in this study, it is also demonstrated that the model formulation can be used to solve the case of DNDP assuming discrete link capacity (discrete number of lanes) for new link additions. For the model formulation, which is still intrinsically nonlinear and noncovex, a global optimization algorithm is developed to solve the model to its exact global optimal solution. Specifically, the original model formulation is firstly relaxed into a mixed integer linear programing problem, whose solution provides the lower bound of the original problem. Then, the lower bound is updated and improved until the global optimization solution is obtained. In constructing the linear programming relaxation, reformulation and linearization technique and mixed-integer outer-approximation approach are adopted. In summary, this paper contributes to the literature in the following aspects: firstly, it provides a novel yet general network design problem formulation to address both the discrete link addition design and continuous road capacity design, which is not studied in previous researches (to our best knowledge). Secondly, a global optimization solution algorithm employing various linearization techniques is developed. Different from the global optimization algorithm used in previous studies (Wang and Lo, 2010 and Luathep et al. 2011), the solution method developed in this study is proved to be able to solve the real global optimum of the original problem, rather than that for only the linearized approximation of the original problem. In addition, the proposed model and solution algorithm could be tailored and adapted to address DNDP with special considerations. For example, the model is shown to be able to solve the network design problem with traffic assignment considering explicit capacity constraints, as well as the DNDP with assumption of discrete capacity levels in design process.

The remainder of this paper is organized as follows. Section 2 presents the original model formulation and the relaxed mixed-integer linear model reformulation. Section 3 proposes the global optimal algorithm. Section 4 discusses several practical considerations. Section 5 reports numerical examples. The final section summarizes the paper.

## 2. Model Formulation

The following notation is used for the formulation.

| | |
|---|---|
| *Sets and parameters* | |
| $A_1$ | Set of existing links in the network |
| $A_2$ | Set of candidate links in the network |
| $A$ | Set of all links in the network, $A = A_1 \bigcup A_2$ |
| $W$ | Set of origin-destination (OD) pairs |
| $d^w$ | Fixed demand between a specified OD pair $w \in W$, $\mathbf{d}^w = [d^w]$ is the vector form demand between the specified OD pair $w$ with a length of $N$ (number of node), wherein the element equals to $d^w$ at the origin node, $-d^w$ at the destination node and 0 otherwise. |
| $\Delta$ | Node-link incidence matrix with a size of $N \times A$, $\Delta = [\delta_a^n]$, where $\delta_a^n = 1$ if node $n$ lies at the entrance of link $a \in A$, $\delta_a^n = -1$ if node $n$ lies at the exit of link $a$, and $\delta_a^n = 0$ otherwise. |
| $\underline{y}_a$ | Lower bound of link capacity for candidate link $a \in A_2$ |
| $\bar{y}_a$ | Upper bound of link capacity for candidate link $a \in A_2$ |
| $B$ | Total available budget |
| $M$ | A large enough positive number |
| $Y_a$ | Link capacity for existing link $a \in A_1$ |
| *Decision variables* | |
| $x_a$ | Continuous link flow variable, $\mathbf{x} = [x_a]$, $a \in A$ |
| $y_a$ | Continuous link capacity variable for candidate link, $\mathbf{y} = [y_a]$, $a \in A_2$ |
| $u_a$ | Binary decision variable, $\mathbf{u} = [u_a]$, $a \in A_2$. It indicates whether a candidate link is added or not for $a \in A_2$: link $a$ is added to the network if $u_a = 1$ and otherwise if $u_a = 0$. For existing link $a \in A_1$, $u_a$ (as is defined to represent $\gamma_a$ in subsection 2.2.1) indicates whether traffic flows on this link is zero or not: no traffic if $u_a = 0$ and $u_a = 1$ otherwise. |
| $v_a^w$ | Continuous disaggregate link flow between OD pair $w \in W$, $\mathbf{v}^w = [v_a^w]$, $a \in A$ |
| $t_a$ | Link travel time function, $a \in A$ |
| $g_a$ | Investment function for candidate link $a \in A_2$ |

The proposed DNDP model aims to provide the transportation network planner simultaneously optimal decisions on both new link additions (binary variables) and new link capacities (continuous variables). It is assumed that the route choice behavior of network users follows the Wardrop's first principle (Wardrop, 1952). In order to minimize the total network travel time costs subject to a given budget, this problem can be represented as following:

[OP: Original Problem]

$$\min_{x,y,u} Z_{\mathbf{OP}} = \sum_{a\in A_1} x_a t_a\left(x_a\right) + \sum_{a\in A_2} x_a t_a\left(x_a, y_a, u_a\right) \tag{1}$$

Subject to:

$$\underline{y}_a \le y_a \le \overline{y}_a, \quad \forall a \in A_2 \tag{2}$$

$$\sum_{a\in A_2} g_a\left(y_a, u_a\right) = \sum_{a\in A_2} \alpha u_a y_a + \beta u_a \le B \tag{3}$$

$$x_a \le u_a M, \quad \forall a \in A_2 \tag{4}$$

$$u_a \in \{0,1\}, \quad \forall a \in A_2 \tag{5}$$

$$\mathbf{x} = \mathbf{x}^*\left(\mathbf{y}, \mathbf{u}\right) \tag{6}$$

$$t_a\left(x_a\right) = T_a\left(1 + R_a\left(\frac{x_a}{Y_a}\right)^4\right), \ \forall a \in A_1 \tag{7}$$

$$t_a\left(x_a, y_a, u_a\right) = T_a\left(1 + R_a\left(\frac{x_a}{y_a}\right)^4\right) + (1-u_a)M, \ \forall a \in A_2 \tag{8}$$

The objective function of this formulation in Eq. (1) is the total travel time cost from both existing links and candidate links. Constraint (2) expresses the restriction of candidate road capacity. Budgetary constraint (3) entails that the total construction cost is less than the maximum allowable expenditure for network improvement. In constraint (3), the second term $\beta u_a$ indicates the fixed cost of new road, that is, the fixed cost $\beta$ is needed once the link is planned to be constructed ( $u_a = 1$ ); a bilinear term i.e., $\alpha u_a y_a$ is used to describe the construction cost: if a candidate link is to be added, i.e., $u_a = 1$, the construction cost is assumed to be a linear function with respect to the link capacity; otherwise, if it is not to be added, i.e., $u_a = 0$, the construction cost will be zero. Constraints (4) and (5)ensure that there is no flow on a link if the link is not constructed, i.e., if $u_a = 0$, then $x_a = 0$. Constraint (6) enforces the flow pattern with Deterministic User Equilibrium (DUE), where $\mathbf{x}^*\left(\mathbf{y}, \mathbf{u}\right)$ is the vector of DUE flows for given vector of link capacities $\mathbf{y}$ and vector of binary decision variables $\mathbf{u}$. Constraints (7) and (8) use the typical BPR function to define the link travel time. In (7), the travel times for existing links only depend on the travel flows $x_a$ as the link capacities $Y_a$ are given. In constraint (8), for candidate link additions, when a candidate link is planned to be constructed, i.e. $u_a = 1$, the additional term $(1-u_a)M$ equals to zero and thus link travel time is described by traditional BPR travel time function; meanwhile, when a link is not to be constructed, i.e., $u_a = 0$, the link travel time will be subject to a big enough constant $M$. The positive and big enough value of travel time for unconstructed link (when $u_a = 0$) as imposed in constraint (8) is to ascertain that no traveler will use this link if it is not even constructed when deterministic user equilibrium principle is applied to capture travelers' routing choice behavior; however, it will not affect the objective function, as constraints (3) and (4) ensure zero traffic flow on unconstructed link and therefore the term $x_a t_a\left(x_a, y_a, u_a\right)$ is still equal to zero.

It should be noted that, in this model formulation, each candidate link is associated with two decision variables, $u_a$ and $y_a$, which combine to describe the link addition plan, whether the link will be constructed or not, and what should be the new link capacity if constructed. If the link capacity $y_a$ is predetermined, this model will reduce to a conventional DNDP; on the other hand, if the link additions $u_a$ are given, this model is indeed a classical CNDP. The obvious nonlinearity property of the model formulation comes from two parts: the bilinear term $u_a y_a$ in constraint (3) and BPR travel time function in constraints (7) and (8). In designing global optimization solution method for this model, different techniques are applied to deal with the two types of nonlinear terms.

2.1. Variational inequality function of traffic assignment problem

As is mentioned in the last section, in this paper, the traffic flow is assumed to be in a pattern of deterministic user equilibrium, i.e., Eq (6), which follows the Wardrop's first principle. Here, the DUE condition is represented by a Variational Inequality (VI) problem (Dafermos, 1980; Smith, 1979). The advantages of VI formulation have been widely recognized: this formulation is only related to link flows, thus avoiding the complicated path enumeration process; more importantly, it can be conveniently used to represent network equilibrium with asymmetric and non-separable travel cost function, i.e., considering interaction between traffic on different roads (Dafermos, 1980). For a given fixed network investment plan $\left(\mathbf{y},\mathbf{u}\right)$, the VI problem is to find the optimal solution $\mathbf{x}^* \in \Psi$ which satisfies the following constraints

$$\sum_{a\in A_1} t_a\left(x_a^*\right)\cdot(x_a^* - x_a) + \sum_{a\in A_2} t_a\left(x_a^*, y_a, u_a\right)\cdot(x_a^* - x_a) \le 0, \quad \forall x_a \in \Psi, \tag{9}$$

$$\Psi = \left\{\mathbf{x} \mid x_a = \sum_{w\in W} v_a^w,\ \Delta\cdot\mathbf{v}^w = \mathbf{d}^w,\ v_a^w \ge 0,\ \forall a\in A,\ w\in W\right\}, \tag{10}$$

where $\Psi$ is a feasible set of traffic flow on the network.

Since all the constraints in $\Psi$ are linear, $\Psi$ is actually a bounded polytope. Let $C$ be the indexes set of corner-points of the polytope and thus it is induced that any point $\mathbf{x} \in \Psi$ can be represented by a convex combination of some corner-points that belong to $C$, that is,

$$\mathbf{x} = \sum_{c \in C} \lambda_c \mathbf{x}^c, \tag{11}$$

$$\sum_{c \in C} \lambda_c = 1, \quad 0 \le \lambda_c \le 1, \quad \forall c \in C, \tag{12}$$

where $\lambda_c$ is the weighted factor of the $c$th corner-point $\mathbf{x}^c$ of the polytope $\Psi$. According to this characteristic of the feasible region, the following proposition can be easily derived.

**Proposition 1** For a given network investment plan $(\mathbf{y}, \mathbf{u})$, $\mathbf{x}^* \in \Psi$ is the optimal solution of the VI problem (9)-(10) if and only if $\mathbf{x}^*$ satisfies the following problem

$$\sum_{a \in A_1} t_a\left(x_a^*\right) \cdot (x_a^* - x_a^c) + \sum_{a \in A_2} t_a\left(x_a^*, y_a, u_a\right) \cdot (x_a^* - x_a^c) \le 0, \quad \forall c \in C \tag{13}$$

Proof. Refer to Luathep et al. (2011).

In conclusion, Eq. (14) can be formulated to stand for the VI problem of the DUE condition.

$$\sum_{a \in A_1} t_a\left(x_a^*\right) \cdot (x_a^* - x_a^c) + \sum_{a \in A_2} [t_a\left(x_a^*, y_a\right) + (1 - u_a) M] \cdot (x_a^* - x_a^c) \le 0, \quad \forall c \in C \tag{14}$$

where $x_a^* \in \Psi$. □

## 2.2. Reformulation of multivariate polynomial function

In this section, we deal with two types of nonlinear terms, i.e., the multivariate link travel time functions and the bilinear functions. The link travel time functions will be reformulated into logrithmic functions, which are univariate and globally concave. Thus, less effort is needed in the process of linearization and relaxation as compared to the multivariate travel time functions as shown in (8). For bilinear functions, the Reformulation-Linearization Technique (RLT) will be applied to transform bilinear functions into equivalent linear constraints.

In the original problem (OP) model, there are two polynomial functions on the list of reformulation, that is, the link travel cost function and the total travel cost function.

### *2.2.1. Link travel time function*

In this paper, the link travel cost function follows the typical Bureau of Public Roads (BPR) equation, which is

$$t_a\left(x_a, y_a\right) = T_a\left(1 + R_a\left(\frac{x_a}{y_a}\right)^4\right), \quad \forall a \in A \tag{15}$$

where $T_a$ is free flow travel cost; both $T_a$ and $R_a$ are given BPR parameters. It should be noted that for existing link $a \in A_1$, $x_a$ is the only variable in the function, whereas for candidate link $a \in A_2$, both $x_a$ and $y_a$ are decision variables.

Let a new variable $h_a$ to represent the monomial $(x_a)^4 / (y_a)^4$ as in (15) we have:

$$h_a = \left( \frac{x_a}{y_a} \right)^4, \ \forall a \in A \tag{16}$$

Since $x_a \geq 0, \forall a \in A$, we cannot take logarithm on both sides of Eq. (16). To solve this issue, two additional nonzero continuous variables $\tilde{x}_a \ (0 < \tilde{x}_a \leq M, \forall a \in A)$ and $\tilde{h}_a \ (0 < \tilde{h}_a \leq M, \forall a \in A)$, and a binary variable $\gamma_a \ (\gamma_a \in \{0,1\}, \ \forall a \in A)$ are introduced for each link $a \in A$. Let

$$x_a = \gamma_a \tilde{x}_a, \quad \forall a \in A \tag{17}$$

$$\tilde{h}_a = \left( \frac{\tilde{x}_a}{y_a} \right)^4, \ \forall a \in A \tag{18}$$

Thus, by substituting Eq. (17) into Eq. (16), the following Eq. (19) can be induced:

$$h_a = \gamma_a \tilde{h}_a, \quad \forall a \in A \tag{19}$$

The binary variable $\gamma_a, a \in A$ is introduced to describe whether link $a$ will be used or not. One can prove that, in the solutions of the OP, $\gamma_a = u_a, a \in A_2$.

**Proposition 2.** Adding constraints $u_a^* = \gamma_a^*, a \in A_2$ into OP will not change the optimal solution of the OP.

Proof. If $u_a^* = 0$, which means link $a$ is not constructed, $\gamma_a^* = x_a^* = 0$ is immediately true due to the constraint (4).

If $\mu_a^* = 1$ and $\gamma_a^* = 1$, which means, in the optimal solution of OP, if a candidate link is constructed, it must be used or $\gamma_a^* = 1$.

If in the optimal solution of OP $\left( x_a^*, y_a^*, u_a^*, \gamma_a^*, h_a^* \right)$, $u_a^* = 1, \gamma_a^* = 0$ for some links, one can always find another optimal solution with the same objective value by only letting $u_a^* = 0$, which will not change the resultant traffic flow pattern $x_a^*$ and the budget constraint will not be violated. In other words, if $u_a^* = 1, \gamma_a^* = 0$ are true in your optimal investment plan, which means a new link addition is completely not used in the network, we can just decide not to construct this new link.

This new investment plan will remain optimal, which will not change the resultant equilibrium traffic pattern and thus the objective value of total network travel time; however, reduce the construction cost, making the budget constraint still fulfilled. That is to say, even in optimal solutions, $u_a^* = 1, \gamma_a^* = 0$, we can obtain an equivalent optimal solution (i.e., with the same objective value) by just letting $u_a^* = 0, \gamma_a^* = 0$. □

Therefore, $\gamma_a, a \in A_2$ is actually the binary investment decision variable $u_a, a \in A_2$ for candidate new links. For an existing road $a \in A_1$, $\gamma_a$ only indicates whether traffic flow on this link is zero or not. To simplify the denotation, we also use the binary variable $u_a$ ($\forall a \in A_1$) to represent $\gamma_a$ of existing links, which results in:

$$x_a = u_a \tilde{x}_a, \quad \forall a \in A \tag{20}$$

$$h_a = u_a \tilde{h}_a, \quad \forall a \in A \tag{21}$$

Taking logarithm on both sides of Eq. (18), we have:

$$\ln \tilde{h}_a = 4 \ln \tilde{x}_a - 4 \ln y_a, \quad \forall a \in A \tag{22}$$

So far, the monomial in the BPR function is transformed into Eq. (22), wherein the nonlinearity is only contained in the logarithmic functions. That is, other than these logarithmic functions, Eq. (22) is in fact in linear form. Let $L_{ha} = \ln\left(\tilde{h}_a\right)$, $L_{xa} = \ln\left(\tilde{x}_a\right)$ and $L_{ya} = \ln\left(y_a\right)$, we have

$$L_{ha} = 4L_{xa} - 4L_{ya}, \quad \forall a \in A \tag{23}$$

The link travel cost function can be replaced by:

$$t_a(x_a, y_a) = T_a + T_a R_a h_a, \quad \forall a \in A \tag{24}$$

The benefits of doing this transformation are apparent: a general nonlinear nonconvex travel time function is now rewritten into several *globally* concave *single-variable* logarithmic functions, which will greatly facilitate the model relaxation in the next section.

*2.2.2. Total system travel cost function*

The total system travel cost also makes use of the BPR equation, the formation of which is quite similar to the link travel time function.

$$\sum_{a \in A} x_a \cdot t_a\left(x_a, y_a\right) = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a \frac{(x_a)^5}{(y_a)^4} \tag{25}$$

Following the same technique introduced in the last subsection, the new variable $p_a$ is used to replace the monomial part in Eq. (25):

$$p_a = \frac{(x_a)^5}{(y_a)^4}, \quad \forall a \in A \tag{26}$$

As is done above, substituting Eq. (20) into Eq. (26) and introducing a new continuous variable $\tilde{p}_a$ $(0 < \tilde{p}_a \leq M, \forall a \in A)$ leads to:

$$\tilde{p}_a = \frac{(\tilde{x}_a)^5}{(y_a)^4}, \quad \forall a \in A \tag{27}$$

$$p_a = u_a \tilde{p}_a, \quad \forall a \in A \tag{28}$$

Taking logarithm on both sides of Eq. (27) leads to:

$$\ln\left(\tilde{p}_a\right) = 5\ln\left(\tilde{x}_a\right) - 4\ln\left(y_a\right), \quad \forall a \in A \tag{29}$$

Similarly, by letting $L_{pa}$ to stand for $\ln\left(\tilde{p}_a\right)$, Eq. (29) is rewritten as

$$L_{pa} = 5L_{xa} - 4L_{ya}, \ \forall a \in A \tag{30}$$

Thus, the total system travel cost can be replaced by:

$$\sum_{a \in A} x_a \cdot t_a\left(x_a, y_a\right) = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a \tag{31}$$

In this case, the objective function can be represented by Eq. (31) because it is exactly the total system travel time.

As for the VI constraints, by plugging Eq. (31) into Eq. (14), we have:

$$\sum_{a \in A} T_a x_a^* + \sum_{a \in A} T_a R_a p_a + \sum_{a \in A_2} (x_a^* - u_a x_a^*) M - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C \tag{32}$$

In Eq. (32), there is one nonlinear term, i.e., $(x_a^* - u_a x_a^*)$. From the proposition 2, we have that the link flow $x_a$ must be positive if a candidate link is planned to be constructed, i.e. $u_a = 1$, whereas apparently there will be no traffic flows on a link if $u_a = 0$. In conclusion, for each candidate link $a \in A_2$, the nonlinear term $(x_a^* - u_a x_a^*)$ is always equal to zero. Then Eq. (32) can be simplified into a linear constraint:

$$\sum_{a \in A} T_a x_a^* + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C \tag{33}$$

It should be noted that, this nonlinear term was relaxed into linear constraints in Luathep et al. (2011), which is indeed deviated from the original constraints.

## 2.3. Linear transformation of bilinear function using RLT technique

In the model formulation, bilinear terms are also involved. In this subsection, we will apply a linearization technique to convert the bilinear terms into equivalent linear constraints, as suggested by Sherali and Adams (1994).

For illustration purpose, this linearization technique is stated as below by taking Eq. (20) $x_a = u_a \tilde{x}_a, \ \forall a \in A$ as an example. It is supposed that $\underline{x}_a \le \tilde{x}_a \le \overline{x}_a$, where $\underline{x}_a$ and $\overline{x}_a$ are respectively a sufficiently small positive number and a sufficiently large upper bound of flow $x_a$. Then, the equivalent linear transformation for each link can be expressed as:

[Linear transformation of bilinear terms]

$$\begin{cases} x_a - u_a \underline{x}_a \ge 0 \\ x_a - u_a \overline{x}_a \le 0 \\ x_a - \tilde{x}_a + \underline{x}_a - u_a \underline{x}_a \le 0 \\ x_a - \tilde{x}_a + \overline{x}_a - u_a \overline{x}_a \ge 0 \end{cases}, \quad \forall a \in A \tag{34}$$

Eq. (20) can be directly rewritten as "if-and-only-if" conditions, which are represented as:

$$\begin{cases} x_a = 0 \Leftrightarrow u_a = 0 \\ x_a = \tilde{x}_a \Leftrightarrow u_a = 1 \end{cases}, \quad \forall a \in A \tag{35}$$

Therefore, by separately substituting two possible values of $u_a$ into Eq. (34), we have:

$$u_a = 0 \Leftrightarrow \left\{ \begin{array}{l} x_a \ge 0 \\ x_a \le 0 \\ x_a - \tilde{x}_a + \underline{x}_a \le 0 \\ x_a - \tilde{x}_a + \overline{x}_a \ge 0 \end{array} \right\} \Leftrightarrow \begin{cases} x_a = 0 \\ \underline{x}_a \le \tilde{x}_a \le \overline{x}_a \end{cases} \tag{36}$$

$$u_a = 1 \Leftrightarrow \left\{ \begin{array}{l} x_a - \underline{x}_a \ge 0 \\ x_a - \overline{x}_a \le 0 \\ x_a - \tilde{x}_a \le 0 \\ x_a - \tilde{x}_a \ge 0 \end{array} \right\} \Leftrightarrow \begin{cases} x_a = \tilde{x}_a \\ \underline{x}_a \le x_a \le \overline{x}_a \end{cases} \tag{37}$$

The above result shows that Eq. (34) is identical to the "if-and-only-if" condition in Eq. (35). Thus, equivalence between Eq. (34) and Eq. (20) is also verified. We can use this linear transformation to replace the bilinear functions in the DNDP model with equivalent linear constraints.

Similarly, given domains of $\tilde{h}_a$ and $\tilde{p}_a$ as defined by $\underline{h}_a \le \tilde{h}_a \le \overline{h}_a$ and $\underline{p}_a \le \tilde{p}_a \le \overline{p}_a$, it is convenient to implement the RLT technique to obtain the equivalent linearization of Eq. (21) and Eq. (28).

It should be noted that the construction cost function $g_a(u_a, y_a)$ also involves a bilinear term in constraint (3).

Considering that the upper bound and lower bound of $y_a$ are already given in the road capacity restriction in Eq. (2), the RLT method can be directly applied to linearize $u_a y_a$. Let $k_a$ to represent the bilinear term $u_a y_a$, we have:

$$g_a(u_a, y_a) = \alpha k_a + \beta u_a, \quad \forall a \in A_2 \tag{38}$$

For simplicity of illustration, let $D$ be a set of variables $D = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\}$; for any variable $d \in D$, $\underline{d}$ and $\overline{d}$ are the lower and upper bounds respectively, and $\hat{d}$ stands for a bilinear term $u_a d$. In summary, the reformulated DNDP problem can be expressed as follows:

[MINLP: Mixed-integer Non-Linear Problem]

$$\min_{\mathbf{x,y,u}} Z_{MNLP} = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a \tag{39}$$

Subject to:

$$\underline{y}_a \le y_a \le \overline{y}_a, \quad \forall a \in A_2 \tag{40}$$

$$\sum_{a \in A_2} \alpha k_a + \beta u_a \le B \tag{41}$$

$$x_a \le u_a M, \quad \forall a \in A \tag{42}$$

$$\sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \le 0, \quad \forall c \in C \tag{43}$$

$$t_a = T_a + T_a R_a h_a, \quad \forall a \in A \tag{44}$$

$$L_{ha} = 4L_{xa} - 4L_{ya}, \quad \forall a \in A \tag{45}$$

$$L_{ha} = \ln(\tilde{h}_a), \quad \forall a \in A \tag{46}$$

$$L_{xa} = \ln(\tilde{x}_a), \quad \forall a \in A \tag{47}$$

$$L_{ya} = \ln(y_a), \quad \forall a \in A \tag{48}$$

$$L_{pa} = 5L_{xa} - 4L_{ya}, \quad \forall a \in A \tag{49}$$

$$L_{pa} = \ln\left(\tilde{p}_a\right), \quad \forall a \in A \tag{50}$$

$$\left.\begin{array}{l} \hat{d} - u_a \underline{d} \geq 0 \\ \hat{d} - u_a \overline{d} \leq 0 \\ \hat{d} - d + \underline{d} - u_a \underline{d} \leq 0 \\ \hat{d} - d + \overline{d} - u_a \overline{d} \geq 0 \\ d \in D = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\} \end{array}\right\} \tag{51}$$

$$u_a \in \{0,1\}, \quad \forall a \in A \tag{52}$$

$$\Psi = \left\{ \mathbf{x} \mid x_a = \sum_{w \in W} v_a^w,\ \Delta \cdot \mathbf{v}^w = \mathbf{d}^w,\ v_a^w \geq 0,\ \forall a \in A,\ w \in W \right\} \tag{53}$$

Where $\mathbf{x}, \mathbf{y}$ and $\mathbf{u}$ are vectors of decision variables $x_a$, $y_a$ and $u_a$ respectively.

## 3. Solution algorithm

### 3.1. Model relaxation

#### *3.1.1. Relaxation of general logarithmic function*

One can find that the nonlinearity of the above shown MINLP only lies in the four logarithmic functions: $\ln\left(\tilde{x}_a\right)$, $\ln\left(y_a\right)$, $\ln\left(\tilde{h}_a\right)$ and $\ln\left(\tilde{p}_a\right)$.In this subsection, a linear relaxation (LR) model for a general logarithmic function is introduced. This model is constructed by using a sequence of outer tangent lines and piecewise linear interpolations. Without loss of generality and for convenience of explanation, the nonlinear function $L_{xa} = \ln\left(\tilde{x}_a\right)$ is taken as an instance to illustrate the linear relaxation process of a logarithmic function.

Suppose the feasible region of $\tilde{x}_a$ is a bounded interval $[\underline{x}_a, \overline{x}_a]$. The interval is divided into $N-1$ small segments by selecting $N-2$ breakpoints between the two endpoints $\underline{x}_a$ and $\overline{x}_a$. The series of breakpoints and two endpoints are denoted by $x_a^n,\ \forall n \in 1, 2..., N$. It should be noted that there is no need to partition the interval into equal segments. As shown in Fig. 1, the linear relaxation of the concave logarithmic function $\ln\left(\tilde{x}_a\right)$ is set to be the region below all tangent lines on each breakpoint and endpoint, and above all chord lines between each pair of consecutive points. In Fig. 1, only two breakpoints are used for demonstration.

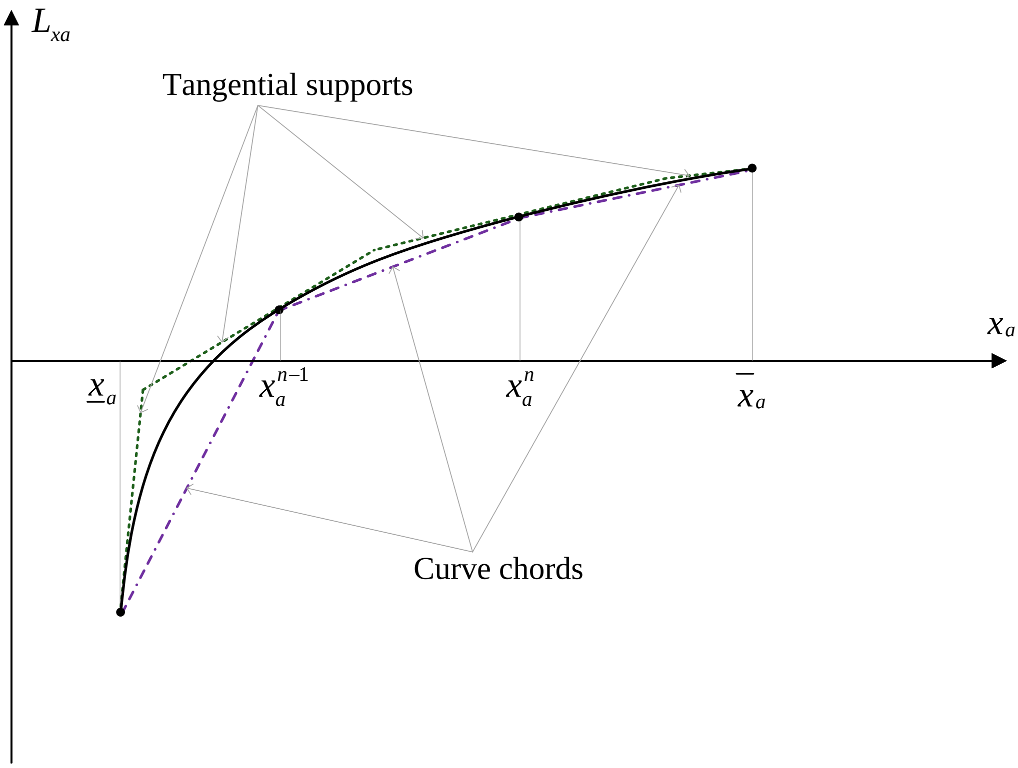


Fig. 1 A linear relaxation for logarithmic function with two breakpoints

The linear relaxation of $\ln\left(\tilde{x}_a\right)$ with breakpoints $\underline{x}_a = x_a^1 \le \ldots \le x_a^n \le \ldots \le x_a^N = \overline{x}_a$ including two endpoints can be specified as follows:

[LR: Linear Relaxation]

$$L_{xa} \le \ln\left(x_a^n\right) - 1 + \frac{x_a}{x_a^n}, \quad \forall x_a^n = \underline{x}_a + \frac{\overline{x}_a - \underline{x}_a}{N-1} \cdot (n-1), \quad n = 1, 2 \ldots, N \tag{54}$$

$$\sum_{n=1}^{N} \theta_{xa}^n x_a^n = x_a \tag{55}$$

$$\sum_{n=1}^{N} \theta_{xa}^n \ln\left(x_a^n\right) \le L_{xa} \tag{56}$$

$$\sum_{n=1}^{N} \theta_{xa}^n = 1 \tag{57}$$

$$\theta_{xa}^n \ge 0, \quad n = 1, 2 \ldots, N \tag{58}$$

$$\theta_{xa}^n \le \lambda_{xa}^{n-1} + \lambda_{xa}^n, \quad n = 2, 3 \ldots, N-1, \quad \theta_{xa}^1 \le \lambda_{xa}^1, \quad \theta_{xa}^N \le \lambda_{xa}^{N-1} \tag{59}$$

$$\sum_{n=1}^{N-1} \lambda_{xa}^{n} = 1 \tag{60}$$

$$\lambda_{xa}^{n} \in \{0,1\}, \quad n = 1,2...,N-1 \tag{61}$$

The upper bound of $\ln(\tilde{x}_a)$ is given in Eq. (54), the right-hand side of which denotes all the tangent lines on each point, whereas the lower bound is provided by Eqs. (55)-(61), which represents all the piecewise linear interpolations between each pair of consecutive points.

To formulate the piecewise linear function, $N$ continuous variables $\theta_{xa}^{n}$, $n = 1,2\ldots,N$ and $N-1$ binary variables $\lambda_{xa}^{n}$, $n = 1,2...,N-1$ are introduced. As shown in Fig. 1, the binary variable $\lambda_{xa}^{n}$ indicates whether an interval is active or not, that is: $x_a$ falls in this interval $[x_a^n, x_a^{n+1}]$ if $\lambda_{xa}^{n} = 1$ and $x_a \notin [x_a^n, x_a^{n+1}]$ if $\lambda_{xa}^{n} = 0$. The continuous variables $\theta_{xa}^{n}$, $n = 1,2\ldots,N$ are the coefficients associated with each breakpoint and measure the location of $x_a$ between the two endpoints of the active interval: specifically, $\theta_{xa}^{n} = \left(x_a - x_a^n\right) / \left(x_a^{n+1} - x_a^n\right)$ and $\theta_{xa}^{n+1} = \left(x_a^{n+1} - x_a\right) / \left(x_a^{n+1} - x_a^n\right)$ if $[x_a^n, x_a^{n+1}]$ is an active interval, whereas the other coefficients of breakpoints are all equal to 0.

Generally, for the case where $x_a$ falls within the active interval $[x_a^n, x_a^{n+1}]$, Eq. (60) guarantees that only $\lambda_{xa}^{n}$ is equal to 1 and all the other $\lambda_{xa}^{m}$, $m = 1...,n-1,n+1...,N-1$ are equal to 0. According to Eqs. (57)-(59), it implies that $\theta_{xa}^{n} + \theta_{xa}^{n+1} = 1$, $\theta_{xa}^{n} \geq 0$, $\theta_{xa}^{n+1} \geq 0$ and $\theta_{xa}^{m} = 0$, $\forall m = 1...,n-1,n+2...,N$. Hence, $x_a$ can be represented by a convex combination from Eq. (55), i.e. $x_a = \theta_{xa}^{n} x_a^n + \theta_{xa}^{n+1} x_a^{n+1}$, and the lower bound of $\ln(\tilde{x}_a)$ can be evaluated from Eq. (56), i.e. $\theta_{xa}^{n} \ln\left(x_a^n\right) + \theta_{xa}^{n+1} \ln\left(x_a^{n+1}\right) \leq L_{xa}$. Combined with Eq. (54), the feasible region $\theta_{xa}^{n} \ln\left(x_a^n\right) + \theta_{xa}^{n+1} \ln\left(x_a^{n+1}\right) \leq L_{xa} \leq \ln\left(x_a^n\right) - 1 + x_a / x_a^n$ is obtained to serve as relaxation of $\ln(\tilde{x}_a)$.

In the above linear relaxation model, the nonlinear function $\ln(\tilde{x}_a)$ is replaced by a set of mixed-integer linear constraints, which serves as its outer approximation. Following the same method, each logarithmic function in the MINLP model can be substituted by a LR programming.

*3.1.2. Relaxation of the DUE condition*

The DUE condition in the MINLP model is formulated as a VI problem related with a set of all corner-points of the traffic flow feasible region $\Psi$. However, the number of VI constraints is extremely large due to the huge number of corner-points, which will notably influence the computation efficiency in solving the MINLP model. Fortunately, because some of the VI constraints are not binding at the optimal solution, a subset of corner-points can be used to define relaxed VI constraints. It is proved that in some conditions a solution to a relaxed VI problem is also the solution to the original VI problem, i.e. the equilibrium traffic flow (Luathep et al., 2011).

Let $C_s$ be a subset of the traffic flow feasible region, i.e. $C_s \subseteq C$, the relaxed VI constraints can be expressed as follows:

$$\sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a - \sum_{a\in A} t_a x_a^c - \sum_{a\in A_2} (1-u_a) M \cdot x_a^c \le 0, \quad \forall c \in C_s, \quad C_s \subseteq C \tag{62}$$

wherein $x_a \in \Psi$. Thus, given a reduced set of corner-points $C_s$, the relaxed MINLP problem can be formulated as:

[R-MINLP: Relaxed MINLP]

$$\min_{x,y,u} Z_{R-MNLP} = \sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a \tag{63}$$

Subject to the same constraints (40)-(42), (44)-(53), and the relaxed VI constraints (62).

The set $C_s$ can be updated iteratively by searching for new corner-points via a linear programming (LP-min). A stopping criteria needs to be satisfied in the iterative process, in which situation it infers that the solution of the relaxed problem R-MINLP also meets the original VI constraints, i.e. it is also the solution of MINLP (see Appendix).

Finally, by further relaxing general logarithmic function in the R-MINLP problem, a relaxed MILP model (denoted as R-MILP) is formulated, which is a linear relaxation of the original problem. Without loss of generality, we let $Q$ stand for the set of variables, whose logarithmic functions need linear relaxation. Thus, for arbitrary variable $q \in Q$, $\underline{q}$ and $\overline{q}$ are the lower and upper bounds respectively; $L_q$ denotes the logarithmic function $\ln(q)$, for example, $L_q$ actually represents $L_{ya}$ if $q = y_a$. In conclusion, the R-MILP can be expressed as follows:

[R-MILP: Relaxed MILP]

$$\min_{x,y,u,t} Z_{R-MILP} = \sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a \tag{64}$$

Subject to:

$$\underline{y}_a \le y_a \le \overline{y}_a, \quad \forall a \in A_2 \tag{65}$$

$$\sum_{a\in A_2} \alpha k_a + \beta u_a \le B \tag{66}$$

$$x_a \le u_a M, \quad \forall a \in A \tag{67}$$

$$\sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a - \sum_{a\in A} t_a x_a^c - \sum_{a\in A_2} (1-u_a) M \cdot x_a^c \le 0, \quad \forall c \in C_s, \quad C_s \subseteq C \tag{68}$$

$$t_a = T_a + T_a R_a h_a, \quad \forall a \in A \tag{69}$$

$$L_{ha} = 4L_{xa} - 4L_{ya}, \quad \forall a \in A \tag{70}$$

$$L_{pa} = 5L_{xa} - 4L_{ya}, \quad \forall a \in A \tag{71}$$

$$\left.\begin{array}{l} \hat{d} - u_a \underline{d} \geq 0 \\ \hat{d} - u_a \overline{d} \leq 0 \\ \hat{d} - d + \underline{d} - u_a \underline{d} \leq 0 \\ \hat{d} - d + \overline{d} - u_a \overline{d} \geq 0 \\ d \in D = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\} \end{array}\right\} \tag{72}$$

$$u_a \in \{0,1\}, \quad \forall a \in A \tag{73}$$

$$\Psi = \left\{ \mathbf{x} \mid x_a = \sum_{w \in W} v_a^w, \ \Delta \cdot \mathbf{v}^w = \mathbf{d}^w, \ v_a^w \geq 0, \ \forall a \in A, \ w \in W \right\} \tag{74}$$

$$\left.\begin{array}{l} L_q \leq \ln\left(q^n\right) - 1 + \dfrac{q}{q^n}, \quad \forall q^n = \underline{q} + \dfrac{\overline{q} - \underline{q}}{N-1} \cdot (n-1), \quad n = 1, 2 \ldots, N \\ \sum_{n=1}^{N} \theta_q^n q^n = q \\ \sum_{n=1}^{N} \theta_q^n \ln\left(q^n\right) \leq L_q \\ \sum_{n=1}^{N} \theta_q^n = 1 \\ \theta_q^n \geq 0, \quad n = 1, 2 \ldots, N \\ \theta_q^n \leq \lambda_q^{n-1} + \lambda_q^n, \quad n = 2, 3 \ldots, N-1, \quad \theta_q^1 \leq \lambda_q^1, \quad \theta_q^N \leq \lambda_q^{N-1} \\ \sum_{n=1}^{N-1} \lambda_q^n = 1 \\ \lambda_q^n \in \{0,1\}, \quad n = 1, 2 \ldots, N-1 \end{array}\right\} \tag{75}$$

$$q \in Q = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\} \tag{76}$$

## 3.2. Global optimization algorithm

In this section, a global optimization algorithm is proposed to solve the problem based on the linear relaxation model R-MILP and a range reduction technique.

### *3.2.1. Range reduction technique*

In the R-MILP model, a number of breakpoints are introduced to relax the logarithmic function into a mixed-integer linear programming. In principle, the relaxation model R-MILP will be much tighter if a larger number of breakpoints are adopted. However, introducing large amount of binary variables with these breakpoints will increase the computational load significantly. Therefore, a range reduction technique is applied, which cuts and reduces the feasible region while ensuring the global optimum not excluded. In this way, with only a few breakpoints to relax the feasible region, the obtained R-MILP model is tighter, and thus by solving which a better lower bound can also be achieved. The range reduction technique is indeed implemented through a series of optimization problems (denoted by RRT problems). Specifically, for an arbitrary variable $x \in \mathrm{X}_{\mathrm{var}}$, where $\mathrm{X}_{\mathrm{var}}$ is the set of variables in the R-MILP model, the RRT problem contains two parts: updating the lower bound of $x$ by solving the RRT-L problem and calculating the new upper bound of $x$ through the RRT-U problem. The RRT problems can be stated as follows:

[RRT-L: Range Reduction Technique for updating Lower bound]

$$\underline{x}^{new} = \text{Min } x \tag{77}$$

subject to:

$$\underline{x}^{old} \le x \le \overline{x}^{old}, \quad \forall x \in \mathrm{X}_{\mathrm{var}} \tag{78}$$

$$Z_{R-MILP} \le \overline{Z}_{MINLP} \tag{79}$$

All the other constraints in the R-MILP model except bound constraints. (80)

[RRT-U: Range Reduction Technique for updating Upper bound]

$$\overline{x}^{new} = \text{Max } x \tag{81}$$

subject to:

$$\underline{x}^{old} \le x \le \overline{x}^{old}, \quad \forall x \in \mathrm{X}_{\mathrm{var}} \tag{82}$$

$$Z_{R-MILP} \le \overline{Z}_{MINLP} \tag{83}$$

All the other constraints in the R-MILP model except bound constraints. (84)

where $\underline{x}^{old}$ and $\overline{x}^{old}$ are respectively the current lower and upper bounds for variable $x \in \mathrm{X}_{\mathrm{var}}$ before update, $\overline{Z}_{MINLP}$ is a known upper bound of the global optimal objective function value of the original MINLP model. $\overline{Z}_{MINLP}$ can be obtained from the objective value of the MINLP problem by feeding a feasible road construction plan into the VI problem and then solving it. Otherwise, one can also utilize a traditional local optimal algorithm to calculate a better upper bound value for $\overline{Z}_{MINLP}$. It is worth noted that the set of variables that need bounds update should be carefully selected, because calculating new bounds also influents the algorithm efficiency.

Considering this, for large size network, reducing range for only a part of variables in $X_{var}$ is recommended. What's more, in order to save computational time, it is recommended to always use the latest feasible range to calculate new bounds.

### *3.2.2. Global optimization solution algorithm*

Based on the above analysis, we develop a global optimization solution algorithm for the model formulation. Basically, the R-MILP is solved to obtain the lower bound of the problem, which is updated and improved by applying range reduction technique, until the gap between lower bound and upper bound fulfils certain requirement.

To explain the solution algorithm more clearly, we show the framework of this solution approach in Fig. 2. Roughly, there are three modules contained in the algorithm. Firstly, an initialization module prepares a group of input parameters for the initial formulation of R-MILP problem. Secondly, R-MILP problem is recurrently updated and solved in each iteration to obtain a lower bound of the problem and then, an upper bound of the problem is calculated by making use of the road construction plan in current solution. A range reduction technique is introduced in the iterative process to reduce the feasible region while guarantee the global optimal solution remaining in the new range. Thirdly, the subset of corner-points is updated, which can make sure that the solution meets the DUE condition. Finally, the algorithm will terminate at the global optimal solution.

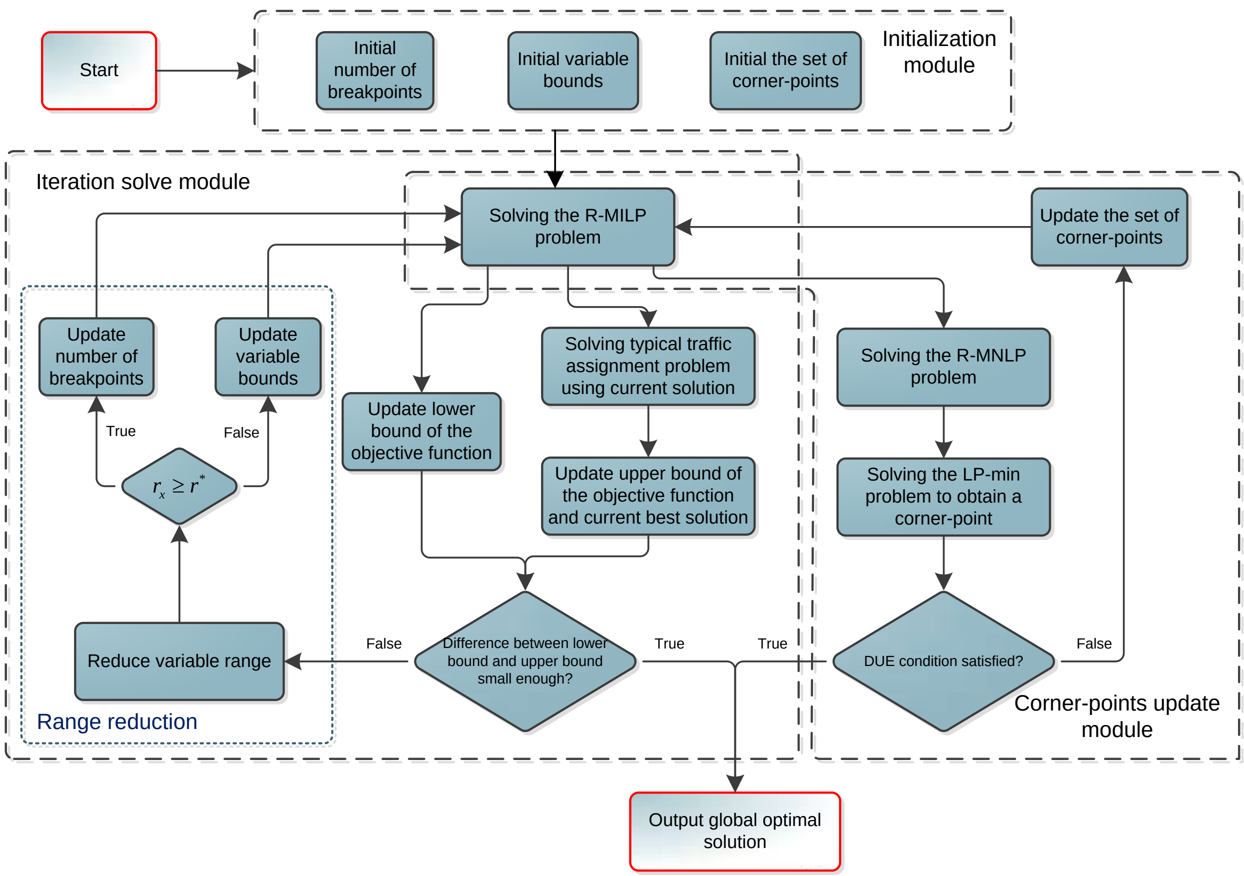


Fig. 2 Framework of algorithm

In summary, the steps of the presented global optimization algorithm can be stated below:

Step 1. Initialization. Use a small integer $n$ as the initial number of break points for each logarithmic curve. Here, $n$ is set to 4. Let the initial bound of $y_a$ be its original domain. For $\tilde{x}_a$, $\tilde{h}_a$, and $\tilde{p}_a$, set a small enough positive numbers as their original lower bounds, and a big enough positive number as their initial upper bounds. Find some corner-points to initialize the set $\{\mathbf{x}^c \mid c \in C_s\}$ to facilitate the formulation of the R-MILP model. Let the iteration number $i = 1$.

Step 2. Solve the relaxed model. Formulate the R-MILP problem with the incumbent range of variables and the current subset of corner-points. Solve the R-MILP problem to its global optimum $\zeta^i = \{\mathbf{x}^\zeta, \mathbf{y}^\zeta, \mathbf{u}^\zeta, \mathbf{h}^\zeta, \mathbf{p}^\zeta, \mathbf{t}^\zeta\}$ by any commercial solver or traditional MILP algorithms. The corresponding objective function value is denoted by $Z^*_{R-MILP,i}$.

Step3. Update the subset of corner-points. Using $\zeta^i$ as the beginning point, solve the R-MINLP problem with fixed construction plan $\mathbf{u}^\zeta$ to obtain a local optimal solution $\sigma^i = \{\mathbf{x}^\sigma, \mathbf{y}^\sigma, \mathbf{u}^\sigma, \mathbf{h}^\sigma, \mathbf{p}^\sigma, \mathbf{t}^\sigma\}$ $(\mathbf{u}^\sigma = \mathbf{u}^\zeta)$ nearby through conventional MNLP methods. Note that the R-MINLP problem only has one group of integer variable $\mathbf{u}$. Once $\mathbf{u}$ is fixed, the R-MINLP problem is reduced to an NLP problem. Formulate the LP-min problem (refer to Appendix) with $\sigma^i$ and solve it to obtain $\mathbf{x}'^*$. Check whether the condition $\sum_{a\in A} T_a x_a^\sigma + \sum_{a\in A} T_a R_a p_a^\sigma - \sum_{a\in A} t_a^\sigma x_a'^* - \sum_{a\in A_2} (1-u_a^\sigma) M \cdot x_a'^* \le \varepsilon$ is true or not. Do nothing if this condition is true or add $\mathbf{x}'^*$ to the subset of corner points for the next iteration otherwise.

Step 4. Update the objective function bounds and check convergence. Substitute $\mathbf{y}^\sigma$ and $\mathbf{u}^\sigma$ into the VI problem and solve it to obtain a feasible objective function value $Z^*_{R-MILP,i}$ of the MINLP problem. The upper bound of the objective function value is then updated via $\overline{Z}_i = \min\left\{\overline{Z}_{i-1}, Z^*_{MINLP,i}\right\}$, whereas the lower bound is updated via $\underline{Z}_i = \max\left\{\underline{Z}_{i-1}, Z^*_{R-MILP,i}\right\}$. The approximated global optimal road construction plan is improved to $\mathbf{y}_i^*$ and $\mathbf{u}_i^*$, which is the local solution of the R-MINLP problem corresponding to incumbent $\overline{Z}_i$. Calculate the relative difference between the upper bound and lower bound $\left|\overline{Z}_i - \underline{Z}_i\right| / \underline{Z}_i$.

Step 5. Reduce feasible range of variables. Calculate new bounds for each variable $x \in \mathrm{X}_{\mathrm{var}}$ by employing the range reduction technique. Renew bounds of variable if its new bounds are tighter than old ones.

Step 6. Renew set of breakpoints for each logarithmic curve. Calculate reserved range rate over the previous variable range via $r_x = (\overline{x}^{new} - \underline{x}^{new}) / (\overline{x}^{old} - \underline{x}^{old})$, $x \in \mathrm{X}_{\mathrm{var}}$. If all $r_x \ge r^*$ $(0 < r^* < 1)$, where $r^*$ is a given rate, increase the number of break points $n = n + \hat{n}$ ($\hat{n}$ is a given positive integer) for the selected $\chi$ variables with the largest interval. For better problem approximation, local solutions from the last $i$ iterations can also be included in the set of breakpoints if the distance between a local solution point and the nearest existing breakpoint is larger than a given gap. Update the iteration number $i = i+1$, go to step 2.

Step 7. Iteration terminates. Stop the iteration if $\left|\overline{Z}_i - \underline{Z}_i\right| / \underline{Z}_i < \tau$ and the condition $\sum_{a\in A} T_a x_a^\sigma + \sum_{a\in A} T_a R_a p_a^\sigma - \sum_{a\in A} t_a^\sigma x_a'^* - \sum_{a\in A_2}(1-u_a^\sigma) M \cdot x_a'^* \le \varepsilon$ is true. List the global optimal solution $\mathbf{y}_i^*$ and $\mathbf{u}_i^*$.

In Step 1, the initial corner-point can be found by solving the VI problem in Eq. (14) with the original network. To improve the computational efficiency, more corner-points that may be binding at the optimal solution can be calculated by repeating the iteration in Step 2 and 3: first solve the R-MILP problem and obtain an optimal construction plan $\mathbf{u}^\zeta$, then solve the R-MINLP problem with this construction plan $\mathbf{u}^\zeta$ as input, search for new corner-point via LP-min and add it to $\{\mathbf{x}^c \mid c \in C_s\}$.

In the following proposition, we prove the convergence of the proposed global optimal algorithm.

**Proposition 3** The proposed algorithm converge to the global optimal solution of the MINLP problem and also the OP problem when the iteration number $i \to \infty$.

Proof. Denote the exact global optimal solution of the MINLP problem by $\hat{\mathbf{y}}$ and $\hat{\mathbf{u}}$ and let $\hat{Z}$ be the corresponding objective function value. Because the linear relaxation problem R-MILP always underestimates the MINLP problem, it holds that the objective function value $Z_{R-MILP,i}^*$ from R-MILP is no larger than $\hat{Z}$. On the other side, feasible value of the MINLP problem $Z_{MINLP,i}^*$ always overestimates $\hat{Z}$. Considering $\underline{Z}_i = \max\left\{\underline{Z}_{i-1}, Z_{R-MILP,i}^*\right\}$, $\overline{Z}_i = \min\left\{\overline{Z}_{i-1}, Z_{MINLP,i}^*\right\}$ and the current best solution $\{\mathbf{x}_i^*, \mathbf{y}_i^*, \mathbf{u}_i^*, \mathbf{h}_i^*, \mathbf{p}_i^*, \mathbf{t}_i^*\}$, we have $\underline{Z}_i \le \hat{Z} \le Z_{MINLP}^*(\mathbf{y}_i^*, \mathbf{u}_i^*) = \overline{Z}_i$. When the iteration number $i \to \infty$, the implementation of Proposition 2 and Proposition 3 in Step 3 guarantees that the final solution satisfies the inequality $\sum_{a\in A} T_a x_{a,i}^* + \sum_{a\in A} T_a R_a p_{a,i}^* - \sum_{a\in A} t_{a,i}^* x_a'^* - \sum_{a\in A_2}(1-u_{a,i}^*) M \cdot x_a'^* \le \varepsilon$, i.e. the DUE condition, where $\varepsilon$ is a given gap tolerance and $x_a'^*$ is from the solution of the LP-min problem. What's more, the combination of range reduction technique in Step 5 and renewing set of breakpoints in Step 6 can always updates the bounds $\underline{Z}_i$ and $\overline{Z}_i$ for $\hat{Z}$. Therefore, we will have $\lim_{m\to\infty} \underline{Z}_i = \hat{Z}$ and $\lim_{m\to\infty} \overline{Z}_i = \lim_{m\to\infty} Z_{MINLP}^*(\mathbf{y}_i^*, \mathbf{u}_i^*) = \hat{Z}$. This proves that the proposed global optimal method converge to the exact global optimal solution of the MINLP model. Since the MINLP model is equivalent to the OP, the algorithmic convergence to the real global optimum of the OP is also proved.□

Remark: It should be noted that the gap cannot be guaranteed to completely vanish in finite number of iterations. However, in practice, usually an accuracy requirement will be given. Thus, the global optimal solution can be efficiently obtained, up to the specific accuracy requirement.

## 4. Practical considerations

The proposed model formulation and solution algorithm can also be tailored and extended to address other practical considerations in discrete transportation network design problems. Hereby, two specific scenarios are illustrated.

### 4.1. Link capacitated traffic assignment problem (CTAP)

Though the traditional DUE model is realistic in distributing traffic in a non-saturated network, the model results in a congested network are far from real observation. Due to the application of link cost function, specifically BPR function, the model may leads to a solution containing over-saturated links, where the traffic flows even exceed their capacities. Considering this, the link capacitated traffic assignment is formulated by including the capacity constraints on link flows in the traditional DUE model to improve the performance of traffic assignment in an over-saturated network. The constraints are shown as follows:

$$\begin{aligned} & x_a \le y_a, \quad \forall a \in A_1 \\ & x_a \le u_a y_a, \quad \forall a \in A_2 \end{aligned} \tag{85}$$

With these capacity constraints, some solution methods are specially proposed for CTAP (Nie et al., 2004). However, the global optimization solution algorithm developed in this study can be immediately applied to solve the DNDP problem even if the capacity constraints are added to the original model. Because we already have $k_a = u_a y_a,\ \forall a \in A_2$, constraints (85) can be rewritten as linear constraints (86) and added to the MINLP and R-MILP model:

$$\begin{aligned} & x_a \le y_a, \quad \forall a \in A_1 \\ & x_a \le k_a, \quad \forall a \in A_2 \end{aligned} \tag{86}$$

Since adding these new linear inequality constraints bring no change to the mathematical property of this problem, the proposed algorithm can still be used and it guarantees the global optimal solution.

### 4.2. Discrete levels of link capacity

In practice, capacity of candidate new road is usually evaluated in discrete number of lanes, which means feasible regions of link capacity variables $y_a$ are discrete, rather than continuous variables. In the above problem formulation, we assume continuous link capacity variables. However, one can find that both the model and the proposed global optimal algorithm can be easily extended to solve the problem with assumption of discrete link capacity.

Suppose link capacity $y_a$ is a discrete variable now and the set $\{1,2,...,m\}$ represents the feasible number of lanes that contained in a candidate link. Thus domain of $y_a$ can be depicted by

$$y_a \in \{y_a^1, y_a^2, ..., y_a^m\}, \quad \forall a \in A_2 \tag{87}$$

Based on this assumption, for MINLP and R-MINLP, we introduce a series of binary variables $\gamma_a^r, r \in \{1,2,...,m\}, a \in A_2$ to indicate whether link capacity $y_a$ is equal to $y_a^r$ or not, i.e. $y_a = y_a^r$ if $\gamma_a^r = 1$ and $y_a \neq y_a^r$ vice versa. In summary, the bounds constraints of $y_a$ in Eq. (40) can be substituted by the following Eq. (88):

$$\left.\begin{array}{l} y_a = \sum_{r=1}^{m} \gamma_a^r y_a^r \\ \sum_{r=1}^{m} \gamma_a^r = 1 \\ \gamma_a^r \in \{0,1\}, \quad \forall r \in \{1,2,...,m\} \\ \forall a \in A_2 \end{array}\right\} \tag{88}$$

Like analysis before, the second and the third constraints in Eq. (88) guarantee there is only one one $\gamma_a^r$ can equal to 1 and all the other $\gamma_a^r$ are forced equal to 0. Thus, from the first constraint, we have $y_a = y_a^r$, only when the associated $\gamma_a^r$ equals to 1.

For the R-MILP model, the LR model cannot be applied immediately in this case because $y_a$ is no longer a continuous variable. Hereby, we remove Eq. (65) and discard $y_a$ from the set $Q$ in Eq. (76). The linear relaxation method for the discrete function $\ln(y_a), a \in A_2$ is amended as below:

$$\sum_{r=1}^{m} \lambda_{ya}^r y_a^r = y_a, \quad y_a^r \in \{y_a^1, y_a^2, ..., y_a^m\} \tag{89}$$

$$\sum_{n=1}^{m} \lambda_{ya}^r \ln(y_a^r) = L_{ya} \tag{90}$$

$$\sum_{r=1}^{m} \lambda_{xa}^r = 1 \tag{91}$$

$$\lambda_{ya}^r \in \{0,1\}, \quad r = 1,2...,m \tag{92}$$

where only a series of binary variables $\lambda_{ya}^r$ are introduced and all the equations are linear. As compared with LR, this model has mainly two differences. First, the weighted factor variables are not needed to approximate the function value between two adjacent feasible points. Second, this model is not a relaxation approximation but provides an exact value of the logarithmic function $\ln(y_a)$.

It is worth noted that, despite the model modification catering for the case with discrete link capacity variables, the essential model properties are not changed, that is, the reformulated R-MILP model remains a mixed-integer linear relaxation of the amended MINLP problem.

Therefore, the proposed global optimal algorithm can still be utilized to solve the problem with discrete levels of link capacity.

## 5. Numerical examples

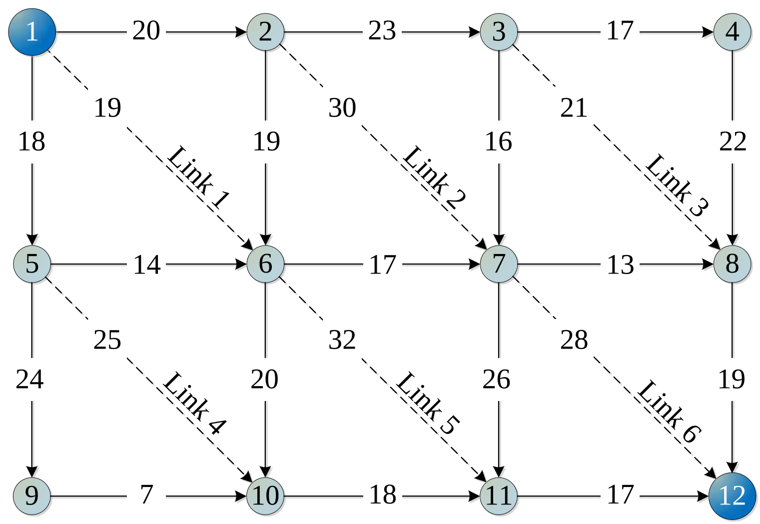


Fig. 3. The 12-node test network.

In this section, to evaluate the model validity and to illustrate the performance of the solution algorithm, a 12-node network, as was used in Gao et al. (2005), is employed as numerical examples. The test network is shown in Fig. 3. It consists of twelve nodes, six candidate links and one O-D pair. Existing links are represented by solid lines and six candidate links by dashed lines. The numbers labeled on these links indicate their free flow travel time $T_a$. The total traffic demand from node 1 to node 12 is supposed to be 20 units. For existing links, $t_a(x_a) = T_a + 0.008x_a^{\ 4}$ is adopted as the travel time function; while for candidate links, the travel time function is assumed to be $t_a\left(x_a, y_a\right) = T_a\left(1 + R_a\left(x_a / y_a\right)^4\right)$, where $R_a = 0.15$ and $y_a \in [4,6]$. The construction cost function $g_a\left(y_a, u_a\right) = \alpha u_a y_a + \beta u_a$ is used in the tests. The value of parameter $\alpha$ and $\beta$ are appropriately set to make the construction cost function value consistent with that in previous studies and given in Table 1. Set $\varepsilon$ in the VI constraints equals to $5\times10^{-5}$. The iteration process terminates if gap between the obtained lower and upper bounds of objective function value is less than $5\times10^{-5}$. The gap tolerance rate is set according to the specific requirement of practical problems. By applying the solution algorithm presented in this study, the global optimization solution of the original problem could be obtained, up to specific accuracy requirement.

All of our tests are run on a personal computer with Intel(R) Xeon(R) CPU E5-2609 0 @ 2.40GHz 2.40GHz (two processors), 32GB RAM and Windows 7 Professional operating system (64-bit). The YALMIP-R20130405 (Löfberg, 2004) together with MATLAB R2012a is used to model all the numerical tests. The commercial optimization solver CPLEX optimization studio 12.3 (IBM ILOG, 2009) is adopted to globally solve all MILP problems, whereas the free solver IPOPT is applied to solve all the nonlinear problems.

Table 1

Value of parameters in construction cost function.

| Candidate link | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $\alpha$ | 1.4200 | 1.0000 | 1.9600 | 2.1200 | 2.7500 | 3.0800 |
| $\beta$ | 0.8308 | 2.1300 | 2.2691 | 2.1356 | 1.3896 | 3.2568 |

5.1. Example 1: comparison between DCNDP and a two-step method.

In this example, results of the proposed model solution are compared with those of a two-step sequential DNDP and CNDP modeling approach. In this two-step sequential method, the traditional DNDP problem with given predetermined road capacities for candidate links is solved first, as was done in Gao et al. (2005) and Luathep et al. (2011). The solution of this DNDP, i.e., the road construction plan, is then applied as the input of the transportation network structure and topology, and a CNDP problem is solved to obtain the optimal road capacity expansion, as was done in Wang and Lo (2010). Table 3 lists the optimal link construction plans with different budgets, wherein results labeled as *'Integrated method (this study)'* are solved by the presented global optimization algorithm in this study. Solutions labeled as *'Two-step method'* are solved from the two-step sequential method. Since the DNDP model needs fixed link capacity and thus fixed link construction cost, in order to have a fair comparison with the integrated method, the parameters shown in Table 2 are adopted in the two-step method. All the other parameters are the same with those used in the integrated method. In columns of *'Capacity'* in Table 3, 1 indicates the corresponding candidate link should be built and 0 otherwise. We calculate objective function value twice in the two-step method: the first time after link addition in step 1 and the second after link expansion in step 2. Both of them are reported in Table 3 and the latter is the final objective function value of the two-step method. In principle, our model can simultaneously provide solution of both candidate links to be constructed and optimal capacities of new links and the solution is global optimal, whereas the sequential two-step method can only optimize one type of variable in each step while assuming the other one is fixed thus the solution is local optimal. Table 3 shows the computational results: the proposed model generally provides better network design plans. The network performance of construction plan from the proposed model is enhanced by up to 9.99% (=[2460.0762-2214.4123]/2460.0762) than that of the two-step method. We also notice that the results from the two methods may be the same, for example, in the cases with given budget of 10, 20 and 60. In summary, the computational results are consistent with the theoretical analysis, that is, the solution of the proposed model in this study may provide a network design plan that is better than simply applying the sequential two-step method, if not the same. One interesting finding that can be observed from Table 3 is that the result with budget 50 is even worse than the result with budget 40 when using the two-step method, which is still because this method can only solve local optimal, and its solution is highly affected by the selection of predetermined road capacities for link additions and cannot guarantee the best network construction plan. This result, from another point of view, justifies the necessity of our integrated model.

On the other side, the proposed solution algorithm for the DCNDP model is globally optimal. The updating process of each iteration with different budgets is shown in Table 4. For each iteration, it presents the evolving upper and lower bounds of objective function value, gap between the two bounds (Gap=[Upper bound-Lower bound]/Upper bound) and solution of relaxed MILP. From this table, one can find that the solution algorithm converges very fast and the global optimal solution can be obtained in a small number of iteration. The computational time for the three cases is 21.7 min, 8.1 min and 34.3 min respectively. In practice, the computational time and the number of iteration may be improved by using different initial set of corner-points, choosing different range reduction variable in set $\mathrm{X}_{\mathrm{var}}$ , rescaling the feasible region of variable and other techniques that can improve the efficiency of MILP. Here, we only set $\mathrm{X}_{\mathrm{var}} = \{x_a, h_a, \forall a \in A; y_a, \forall a \in A_2\}$ and rescale the feasible region of $x_a, \forall a \in A$ and $y_a, \forall a \in A_2$ . It should be noted that the number of iteration needed seems not to be related to the value of budget with our method. However, in Gao et al. (2005), the budget value affects the required number of iteration significantly, and larger number of iterations is needed with their solution method.

Table 2

Parameters adopted in two-step method.

| DNDP parameters | | | | | | | CNDP parameters |
|---|---|---|---|---|---|---|---|
| Candidate link | 1 | 2 | 3 | 4 | 5 | 6 | Construction cost |
| Link capacity | 4 | 4 | 4 | 4 | 4 | 4 | $g_a(y_a) = \alpha(y_a - 4)$ |
| Construction cost | 6.5108 | 6.1300 | 10.1091 | 10.6156 | 12.3896 | 15.5768 | |

Table 3

Optimal link construction results for the 12-node network.

| Budget | Two-step method | | | | Integrated method (this study) | | | Network performance enhancement |
|---|---|---|---|---|---|---|---|---|
| | Step1: DNDP | | Step2: CNDP | | | | | |
| | New links | Objective value | Capacity | Objective value | New links | Capacity | Objective value | |
| 10 | 1 0 0 0 0 0 | 4117.6890 | (6) | 3959.2197 | 1 0 0 0 0 0 | (6) | 3959.2197 | 0.00% |
| 20 | 1 0 0 0 1 0 | 3875.4668 | (4.7744, 4) | 3795.5799 | 1 0 0 0 1 0 | (4.7744, 4) | 3795.5799 | 0.00% |
| 30 | 1 1 0 0 0 1 | 2678.0491 | (4.8844, 4, 4.1710) | 2584.8569 | 1 0 0 0 0 1 | (6, 5.6469) | 2488.8950 | 3.71% |
| 40 | 1 1 1 0 0 1 | 2549.8698 | (4.7736, 4, 4, 4.1866) | 2459.8584 | 1 1 0 0 0 1 | (6, 6, 6) | 2315.5778 | 5.87% |
| 50 | 1 1 1 1 0 1 | 2523.1283 | (4.6963, 4, 4, 4, 4.0224) | 2460.0762 | 1 1 0 0 1 1 | (6, 4, 4.1428, 6) | 2214.4123 | 9.99% |
| 60 | 1 1 1 0 1 1 | 2406.3574 | (6, 4.2837, 4, 4, 6) | 2123.1311 | 1 1 1 0 1 1 | (6, 4.2837, 4, 4, 6) | 2123.1311 | 0.00% |
| 70 | 1 1 1 1 1 1 | 2383.1818 | (6, 4, 4, 4, 4, 5.8922) | 2108.9950 | 1 1 1 0 1 1 | (6, 6, 5.4203, 6, 6) | 2104.3019 | 0.22% |

Table 4

Progression of iteration results with different budgets.

| Budget | Lower bound | Upper bound | Gap | New links | Capacity |
|---|---|---|---|---|---|
| 20 | | 5500.9609 | | | |
| | 1563.4780 | 3932.9888 | 60.2471% | 1 0 0 1 0 0 | (6.0000, 4.0159) |
| | 3029.1426 | 3850.0410 | 21.3218% | 1 0 1 0 0 0 | (5.8428, 4.3895) |
| | 3437.4043 | 3799.5409 | 9.5311% | 1 0 0 0 1 0 | (4.7171, 4.0296) |
| | 3609.2963 | 3795.5799 | 4.9079% | 1 0 0 0 1 0 | (4.7744, 4.0000) |
| | 3795.5444 | 3795.5799 | 0.0009% | 1 0 0 0 1 0 | (4.7744, 4.0000) |
| | Result: | 3795.5799 | Iteration number: | 5 | Computational time: 21.7 min |
| 40 | | 5500.9609 | | | |
| | 1561.2002 | 2315.5778 | 32.5784% | 1 1 0 0 0 1 | (6.0000, 6.0000, 6.0000) |
| | 1644.3193 | 2315.5778 | 28.9888% | 1 1 0 0 0 1 | (6.0000, 6.0000, 6.0000) |
| | 2122.6644 | 2315.5778 | 8.3311% | 1 1 0 0 0 1 | (6.0000, 6.0000, 6.0000) |
| | 2245.6877 | 2315.5778 | 3.0183% | 1 1 0 0 0 1 | (6.0000, 6.0000, 6.0000) |
| | 2315.5159 | 2315.5778 | 0.0027% | 1 1 0 0 0 1 | (6.0000, 6.0000, 6.0000) |
| | Result: | 2315.5778 | Iteration number: | 5 | Computational time: 8.1 min |
| 60 | | 5500.9609 | | | |
| | 1561.1886 | 2182.7586 | 28.4764% | 1 1 0 0 1 1 | (6.0000, 6.0000, 4.0776, 6.0000) |
| | 1643.1247 | 2182.7586 | 24.7226% | 1 1 0 0 1 1 | (6.0000, 6.0000, 4.0776, 6.0000) |
| | 2027.3339 | 2170.4235 | 6.5927% | 1 1 0 0 1 1 | (6.0000, 6.0000, 6.0000, 6.0000) |
| | 2075.4905 | 2170.4235 | 4.3739% | 1 1 0 0 1 1 | (6.0000, 6.0000, 6.0000, 6.0000) |
| | 2121.4478 | 2123.1311 | 0.0793% | 1 1 1 0 1 1 | (6.0000, 4.2837, 4.0000, 4.0000, 6.0000) |
| | 2123.1269 | 2123.1311 | 0.0002% | 1 1 1 0 1 1 | (6.0000, 4.2837, 4.0000, 4.0000, 6.0000) |
| | Result: | 2123.1311 | Iteration number: | 6 | Computational time: 34.3 min |

## 5.2. Example 2: DCNDP with discrete levels of capacity improvements

We also test the DCNDP model with discrete levels of capacity improvements. The test network is identical, i.e., the 12-node network shown in Fig. 3. It is assumed that $y_a \in \{4,5,6\}, \forall a \in A$ and all the other value of parameters are the same as those used in example 1. The results of this test with different budgets are exhibited in Table 5. One can find the optimal objective value is no better than the results of DCNDP as shown in Table 3 with continuous capacity enhancement for new link additions, which can be easily interpreted by the more stringent constraint of discrete link capacity.

Table 5

Optimal solutions of DCNDP model with discrete levels of capacity improvements.

| Budget | New links | Capacities | Objective value from MILP | Exact objective value | Gap | Iteration |
|---|---|---|---|---|---|---|
| 10 | 1 0 0 0 0 0 | (6) | 3959.1757 | 3959.2197 | 0.00111% | 5 |
| 20 | 1 0 1 0 0 0 | (6, 4) | 3865.9238 | 3866.0809 | 0.00406% | 7 |
| 30 | 1 0 0 0 0 1 | (5, 6) | 2526.6700 | 2526.6708 | 0.00003% | 5 |
| 40 | 1 1 0 0 0 1 | (6, 6, 6) | 2315.5769 | 2315.5778 | 0.00004% | 5 |
| 50 | 1 1 0 0 1 1 | (6, 4, 4, 6) | 2216.0031 | 2216.0344 | 0.00141% | 5 |
| 60 | 1 1 1 0 1 1 | (6, 4, 4, 4, 6) | 2125.1241 | 2125.1245 | 0.00002% | 5 |
| 70 | 1 1 1 0 1 1 | (6, 6, 5, 6, 6) | 2104.5681 | 2104.5692 | 0.00005% | 6 |

## 6. Conclusion

In this study, we develop a novel and general discrete network design model formulation and its global optimal solution algorithm to determine the optimal link addition and link capacity construction plan in transportation networks. The model relaxes the assumption that the link capacity for candidate link addition is predetermined and given and treats it as a simultaneous decision variable, which provides a more general transport network design model. Besides, the global optimization solution algorithm is developed based on RLT technique, outer-approximation approach and range reduction technique. Numerical tests are implemented to demonstrate the performance of the proposed model and the solution quality of the algorithm. However, we have to admit that, at current stage, most global optimal solution algorithms are not as efficient as local optimal solution methods, and therefore unattractive especially for practical use. However, it also should be noted that only this type of solution algorithm can guarantee true global optimal solution of the developed model, thus deserves more attention in the future study.

## Acknowledgements

This study is sponsored by the Singapore Ministry of Education AcRF Tier 1 Grant RG117/14, M401030000. The helpful and constructive comments of the anonymous referees are gratefully acknowledged.

## Appendix

Suppose $F=(x_a,t_a,u_a,p_a)$ is the solution to the relaxed VI problem Eq. (62), thus $F$ also meets the original VI constraints Eq. (43), i.e., for arbitrary link flows $x'_a \in \Psi$, the following inequation (A.1) is satisfied.

$$\sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a - \sum_{a\in A} t_a x'_a - \sum_{a\in A_2} (1-u_a) M \cdot x'_a \le 0 . \tag{A.1}$$

In order to judge whether $F$ is the solution to the original VI problem and find new corner-points, Luathep et al. (2011) proposed a optimization-based method, which is briefly stated here.

Significantly, constraint (A.1) equals to following inequation:

$$\max_{x'_a\in\Psi}\left( \sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a - \sum_{a\in A} t_a x'_a - \sum_{a\in A_2} (1-u_a) M \cdot x'_a \right) \le 0 . \tag{A.2}$$

Since $F$ is known, the first two terms of the left-hand side equation can be treated as constants. Thus, we have,

$$\min_{x'_a\in\Psi}\left( \sum_{a\in A} t_a x'_a + \sum_{a\in A_2} (1-u_a) M \cdot x'_a \right) \ge 0 . \tag{A.3}$$

That is, if the minimum value of the multinomial is larger than 0, $F$ is the solution to the original VI. Hence, an unconstrained linear programming is proposed to find the minimum value of the multinomial, which is expressed as:

[LP-min]

$$\min_{x'_a\in\Psi} Z_{LP-\min} = \sum_{a\in A} t_a x'_a + \sum_{a\in A_2} (1-u_a) M \cdot x'_a . \tag{A.4}$$

One can easily solve this LP-min problem to its optimum $x'^{*}_a$ by any traditional algorithm, which is also its global optimal solution considering the global optimality characteristic of linear programming. Therefore, the condition (A.1) is satisfied for all feasible link flows $x'_a \in \Psi$, that is, $F$ is also the solution to the original VI, if $\sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a - \sum_{a\in A} t_a x'^{*}_a - \sum_{a\in A_2} (1-u_a) M \cdot x'^{*}_a \le 0$. Obviously, otherwise if $\sum_{a\in A} T_a x_a + \sum_{a\in A} T_a R_a p_a - \sum_{a\in A} t_a x'^{*}_a - \sum_{a\in A_2} (1-u_a) M \cdot x'^{*}_a > 0$, $F$ is infeasible for the original VI constraints. In this case, $x'^{*}_a$ can be added to the set of corner-points because the solution of the linear programming is always a corner-point of the feasible region $\Psi$.

## References:


Dafermos, S., 1980. Traffic equilibrium and variational inequalities. Transportation Science 14 (1), 42-54.

Drezner, Z., Wesolowsky, G.O., 2003. Network design: Selection and design of links and facility location. Transportation Research Part A 37 (3), 241-256.

Farvaresh, H., Sepehri, M.M., 2011. A single-level mixed integer linear formulation for a bi-level discrete network design problem. Transportation Research Part E 47 (5), 623-640.

Fontaine, P., Minner, S., 2014. Benders decomposition for discrete–continuous linear bilevel problems with application to traffic network design. Transportation Research Part B 70, 163-172.

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.

IBM ILOG, 2009. User's manual for cplex, New York.

Kim, B.J., Kim, W., 2006. An equilibrium network design model with a social cost function for multimodal networks. The Annals of Regional Science 40 (3), 473-491.

Leblanc, L.J., 1975. An algorithm for the discrete network design problem. Transportation Science 9 (3), 183-199.

Li, C., Yang, H., Zhu, D., Meng, Q., 2012. A global optimization method for continuous network design problems. Transportation Research Part B 46 (9), 1144-1158.

Liu, H., Wang, D.Z.W., 2015. Global optimization method for network design problem with stochastic user equilibrium. Transportation Research Part B 72, 20-39.

Löfberg, J., 2004. Yalmip : A toolbox for modeling and optimization in matlab, Proceedings of the 2004 IEEE International Symposium on Computer Aided Control Systems Design, Taipei, Taiwan, pp. 284-289.

Luathep, P., Sumalee, A., Lam, W.H.K., Li, Z.-C., Lo, H.K., 2011. Global optimization method for mixed transportation network design problem: A mixed-integer linear programming approach. Transportation Research Part B 45 (5), 808-827.

Magnanti, T.L., Wong, R.T., 1984. Network design and transportation planning: Models and algorithms. Transportation Science 18 (1), 1-55.

Meng, Q., Yang, H., Bell, M.G.H., 2001. An equivalent continuously differentiable model and a locally convergent algorithm for the continuous network design problem. Transportation Research Part B 35 (1), 83-105.

Nie, Y., Zhang, H., Lee, D.-H., 2004. Models and algorithms for the traffic assignment problem with link capacity constraints. Transportation Research Part B 38 (4), 285-312.

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.

Sherali, H.D., Adams, W.P., 1994. A hierarchy of relaxations and convex hull characterizations for mixed-integer zero—one programming problems. Discrete Applied Mathematics 52 (1), 83-106.

Sherali, H.D., Adams, W.P., 1998. A reformulation-linearization technique for solving discrete and continuous nonconvex problems. Springer.

Smith, M.J., 1979. The existence, uniqueness and stability of traffic equilibria. Transportation Research Part B 13 (4), 295-304.

Solanki, R.S., Gorti, J.K., Southworth, F., 1998. Using decomposition in large-scale highway network design with a quasi-optimization heuristic. Transportation Research Part B 32 (2), 127-140.

Szeto W. Y., Jiang Y., Wang, D. Z. W., Sumalee, A. 2014. A Sustainable Road Network Design Problem with Land Use Transportation Interaction over Time. Network and Spatial Economics. In Press. DOI: 10.1007/s11067-013-9191-9.

Wang, D.Z.W., Lo, H.K., 2010. Global optimum of the linearized network design problem with equilibrium flows. Transportation Research Part B 44 (4), 482-492.

Wang, S., Meng, Q., Yang, H., 2013. Global optimization methods for the discrete network design problem. Transportation Research Part B 50 , 42-60.

Wardrop, J.G., 1952. Some theoretical aspects of road traffic research, Proceedings of the Institution of Civil Engineers, pp. Part II, Vol. 1, 325-378.

Yang, H., Bell, M.G.H., 1998. Models and algorithms for road network design: A review and some new developments. Transport Reviews 18 (3), 257-278.