Global optimization methods for urban road network design problems

Liu, Haoxiang

2016

Liu, H. (2016). Global optimization methods for urban road network design problems.
Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/67035>

<https://doi.org/10.32657/10356/67035>

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

**GLOBAL OPTIMIZATION METHODS FOR
URBAN ROAD NETWORK DESIGN
PROBLEMS**

LIU HAOXIANG

**SCHOOL OF CIVIL AND ENVIRONMENTAL
ENGINEERING**

2016

**GLOBAL OPTIMIZATION METHODS FOR
URBAN ROAD NETWORK DESIGN
PROBLEMS**

LIU HAOXIANG

School of Civil and Environmental Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

2016

To my beloved parents

献给我最亲爱的父亲母亲

ACKNOWLEDGEMENTS

Foremost, I would like to give my sincerest appreciation for my supervisor Dr. Wang Zhiwei. He provided me patient guidance and continuous encouragement. Without his efforts, this thesis cannot be finished.

I am grateful to my former supervisor Professor Gao Ziyou in Beijing Jiaotong University. Without his inspiration, motivation and encouragement, I would not proceed my study in transportation research area.

I also wish to express my sincere thanks to all the committee members of my Qualification Examination for their time, efforts and insightful comments. I learn a lot through their questions and advices.

My special thanks also go to technicians in Geotechnics, Transport and Geospatial Laboratory for their hearty assistance as well as financial support from Nanyang Technological University during my Ph.D. study.

I would like to thank all my colleagues in our research room and friends, in particular Ou Jiajue, Qiu Jishen, Liu Zhidan, Huang Ying, Ashish Nayan, Lei Jiawei, Ban Xu and Shan Xin, for their concern, support and unforgettable friendship. Also I thank my friends Wang Li, Zhang Xin and Fan Wu for their companionship during my stay in Singapore. I would like to show my appreciation to my seniors: Dr. Xie Dongfan, Dr. Li Feng, and Dr. Du Bo for their assistance.

Last but not least, I am extremely grateful to my parents for their love, care, unceasing inspiration and encouragement throughout my life.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	i
TABLE OF CONTENTS.....	iii
LIST OF FIGURES	ix
LIST OF TABLES	xi
LIST OF ABBREVIATIONS	xiii
ABSTRACT.....	xv
Chapter 1 INTRODUCTION.....	1
1.1 Background	1
1.2 Objectives of the research	3
1.3 Outline of this thesis	5
Chapter 2 LITERATURE REVIEW	9
2.1 Continuous network design problem	10
2.2 Discrete network design problem	12
2.3 Concluding remarks	14
Chapter 3 GLOBAL OPTIMIZATION METHOD FOR NETWORK DESIGN PROBLEM WITH STOCHASTIC USER EQUILIBRIUM.....	15
3.1 Introduction	15
3.2 Problem formulation	18
3.2.1 Model formulation	19
3.2.2 Model reformulation with logarithmic functions.....	20

3.3 Solution Algorithm	24
3.3.1 Outer approximation based linear programming relaxation.....	24
3.3.2 Global optimization algorithm for CNDP applying range reduction technique	32
3.4 Computational results	38
3.4.1 Solution quality	43
3.4.2 Effect of the initial feasible objective value	51
3.4.3 Approximate CNDP with DUE problem by adopting a large θ value	53
3.5 Concluding remarks	57
Chapter 4 MODELING AND SOLVING DISCRETE NETWORK DESIGN PROBLEM WITH STOCHASTIC USER EQUILIBRIUM.....	59
4.1 Introduction	60
4.2 Model formulation	63
4.3 Global optimization solution method.....	66
4.3.1 Model transformation and relaxation	67
4.3.2 Algorithm 1: a branch-and-bound based global optimal solution algorithm	75
4.3.3 Algorithm 2: an iterative global optimal solution algorithm based on adding extra constraints	78
4.4 Numerical examples.....	84
4.4.1 Example 1: Simple twelve-node network.....	84
4.4.2 Example 2: Sioux-Falls network.....	87

4.5 Concluding remarks	90
Chapter 5 A NOVEL DISCRETE NETWORK DESIGN PROBLEM FORMULATION AND ITS GLOBAL OPTIMIZATION SOLUTION ALGORITHM.....	93
5.1 Introduction	93
5.2 Model formulation	96
5.2.1 Variational inequality function of traffic assignment problem.....	100
5.2.2 Reformulation of multivariate polynomial function	102
5.2.3 Linear transformation of bilinear function using RLT technique...	107
5.3 Solution algorithm	110
5.3.1 Model relaxation	110
5.3.2 Global optimization algorithm.....	115
5.4 Practical considerations.....	122
5.4.1 Link capacitated traffic assignment problem (CTAP)	123
5.4.2 Discrete levels of link capacity	123
5.5 Numerical examples.....	126
5.5.1 Example 1: comparison of DCNDP and a two-step method.	127
5.5.2 Example 2: DCNDP with discrete levels of capacity improvements	132
5.6 Concluding remarks	132
Chapter 6 GLOBAL OPTIMIZATION FOR TRANSPORT NETWORK EXPANSION AND SIGNAL SETTING.....	135
6.1 Introduction	135

6.2 Model formulation	137
6.2.1 Bi-level model for CNDSSP.....	139
6.2.2 Single-level MIP for CNDSSP	141
6.3 Linearization of the single-level model	146
6.3.1 Linearization of two-dimensional functions.....	146
6.3.2 Linearization of objective function and storage capacity function.	154
6.3.3 The reformulated MILP model.....	155
6.3.4 Adapting the MILP for discrete link capacity improvement.....	158
6.4 Numerical studies	159
6.4.1 Case 1: the Yang-Yagar network.....	160
6.4.2 Case 2: the Nguyen-Dupuis network	166
6.5 Concluding remarks	168
Chapter 7 CONCLUSIONS AND FUTURE WORK.....	171
7.1 Thesis summary.....	171
7.2 Recommendation for future work.....	174
Appendix A: A simple example of outer approximation based linear relaxation	177
Appendix B: Data for Sioux-Falls network in Chapter 3	179
Appendix C: Data for Sioux-Falls network in Chapter 4	181
Appendix D: Updating approximation of VI constraint.....	183
Appendix E: A simple example of branching scheme for equivalent J_1 triangulation	185
PUBLICATIONS ARISING FROM THE THESIS	187

REFERENCES.....	189

LIST OF FIGURES

Figure 1.1 Outline of the thesis.	6
Figure 3.1 Linear relaxation for logarithmic function.	25
Figure 3.2 Methods of reducing error of the linear relaxation.	33
Figure 3.3 Seven-link test network.	41
Figure 3.4 14-link test network.	41
Figure 3.5 Sioux-Falls network.	43
Figure 3.6 Update process of upper and lower bounds of the global optimal objective value with different initial feasible objective value on seven-link network.	52
Figure 3.7 Update process of upper and lower bounds of the global optimal objective value when adopting different initial feasible objective value on 14- link network.	53
Figure 4.1 Polyhedral outer-approximation for $\ln(p_r^w)$	71
Figure 4.2 An illustration example for Algorithm 2.	81
Figure 4.3 The twelve-node network.	85
Figure 4.4 The binary tree of the proposed Algorithm 1 when budget=20.	88
Figure 4.5 Sioux-Falls network.	89
Figure 5.1 Framework of algorithm.	118
Figure 5.2 The 12-node test network.	126
Figure 6.1 An equivalent J_1 triangulation of the feasible region	149
Figure 6.2 Different piecewise-linear approximations for $f(x)$	150
Figure 6.3 The small Yang-Yagar network.	160
Figure 6.4 The Nguyen-Dupuis network.	166

LIST OF TABLES

Table 3.1 Comparison of the test network size to those in the literature.....	40
Table 3.2 Data for the seven-link test network.	41
Table 3.3 Data for the 14-link test network.	42
Table 3.4 Results of the seven-link network by the proposed global optimal algorithm.	44
Table 3.5 Results of the seven-link network solved by the method of Wang and Lo (2010) in different partition schemes.	45
Table 3.6 Results of the 14-link network by the proposed global optimal algorithm.	46
Table 3.7 Remaining percentage of feasible range of link flows in each iteration.	48
Table 3.8 Remaining percentage of feasible range of link capacities in each iteration.	48
Table 3.9 Remaining percentage of feasible range of p_r^w in each iteration.....	49
Table 3.10 Results of the Sioux-Falls network via the proposed optimal algorithm.	50
Table 3.11 Abbreviations of solution algorithms and references for solving CNDP with DUE problem.	54
Table 3.12 Model results for approximated CNDP with DUE.	55
Table 4.1 Results of the test network under different budgets by the two proposed global optimal algorithms.	88
Table 4.2 Iterative process of the proposed Algorithm 2.	89
Table 4.3 Results of Sioux-Falls network with three different θ value adopted in logit model.	90
Table 5.1 Value of parameters in construction cost function.	127

Table 5.2 Parameters adopted in two-step method.....	130
Table 5.3 Optimal link construction results for the 12-node network.....	130
Table 5.4 Progression of iteration results with different budgets.	131
Table 5.5 Optimal solutions of DCNDP model with discrete levels of capacity improvements.	132
Table 6.1 Link data input for the small network.	161
Table 6.2 Other parameters used in the network.	161
Table 6.3 Comparison of CNDSSP with other models.	163
Table 6.4 Results of the CNDSSP with discrete capacity enhancement applying the Log-MILP method.	165
Table 6.5 Link parameters for the Nguyen-Dupuis network.	166
Table 6.6 Demand and maximum storage capacities of signal-controlled links of the Nguyen-Dupuis network.	167
Table 6.7 Results generated from the Log-MILP method with continues and discrete capacity enhancements.	169

LIST OF ABBREVIATIONS

BPR	Bureau of Public Roads
CNDP	continuous network design problem
CNDSSP	combined network design and signal setting problem
CTAP	capacitated traffic assignment problem
DCNDP	integrated discrete and continuous network design problem
DLog	logarithmic disaggregated convex combination model
DNDP	discrete network design problem
DUE	deterministic user equilibrium
ENSS	combined traffic assignment and signal control problem
KKT condition	Karush-Kuhn-Tucker condition
Log	logarithmic branching convex combination model
MILP	mixed-integer linear programming
MINLP	mixed-integer nonlinear programming
MIP	mixed-integer programming
MNDP	mixed network design problem
MSA	method of successive averages
NDP	network design problem
NP-hard	non-deterministic polynomial-time hard
O-D	origin-destination
RLT	reformulation-linearization technique
SO	system optimum
SUE	stochastic user equilibrium
VI	variational inequality

ABSTRACT

Adequate infrastructure is required to alleviate the traffic congestion with the sustained growth in demand of passenger and freight transportation. The main study area in this thesis is the related transportation network design problem (NDP), which involves the optimal decision on the expansion of the existing links and/or on the addition of new links within a given budget aiming to achieve the best network performance. Since this class of problem can also be generalized to handle other transport management problem, like road pricing and signal control, or even be applied on other transportation modes, like public transport and shipping, hence, NDP deserves more attention and further studies on its modeling and solution algorithm development issues.

In the literature, NDP problem is often formulated as bi-level programming, which is nonconvex and nonlinear in nature. The solution of nonlinear and nonconvex problem is well recognized to be extremely hard. In the literature, many researchers focused on developing various algorithmic methods for solving NDP, ranging from classic solution algorithms to widely used meta-heuristic methods. However, though a lot of efforts have been made, challenges still exist mainly in two aspects. First, solution accuracy should be enhanced. Due to nonconvex property of the NDP problems, most algorithms developed in previous studies can only find local optimal solution, rather than global optimization solution. Although these solution algorithms are efficient and fast to obtain a “good” solution, the solution quality is compromised, being a local optimal, rather than the globally best. Other than the local optimization method, a class of stochastic global optimization method, including Genetic Algorithm (GA) and simulated annealing (SA), is applied to solve NDP. However, due to

the inherent characteristic of stochastic optimization method, global solution cannot be guaranteed. Second, computational efficiency in terms of calculation time should be improved. NDP problems with real-size network or a large size of candidate projects remain far from being tractable.

The thesis contributes to the NDP problems in two main aspects: firstly, to develop more realistic mathematical programming model for urban transportation network design problems; secondly, to propose efficient global optimization solution algorithms for network design problems and ensure the true optimal transportation planning.

Firstly, a continuous network design problem (CNDP) that aims to optimize the network performance via road capacity expansion is considered in the thesis. Road users are assumed to follow the traffic assignment principle of stochastic user equilibrium, specifically, the logit route choice model, which is a more general model and includes deterministic user equilibrium flow pattern as an extreme case. To obtain the exact global optimal solution of the problem, a global optimal solution algorithm is proposed based on a tight linear programming relaxation. The developed model, which is bi-level and inherently nonlinear and nonconvex, is firstly transformed into a nonlinear programming with only logarithmic functions as nonlinear terms. Then, the tight linear programming relaxation is derived by applying an outer-approximation technique and embedded within a global optimization solution algorithm, which is proved to converge to a global optimum of the target problem. This thesis also applies range reduction technique in the presented algorithm to improve the computational efficiency. This thesis fills the research gap that no global optimal solution method was developed for CNDP with SUE and guarantees exact global optimal solution of the problem to be solved.

The thesis then addresses a discrete network design problem (DNDP) with stochastic user equilibrium constraint that aims to optimize the network performance via new link construction. In order to obtain the global optimal solution of the developed discrete network design problem with stochastic user equilibrium model, which is inherently nonconvex, the original problem is relaxed to a convex mixed-integer nonlinear problem, whose solution provides a lower bound of the original problem. Two iterative global optimization solution methods with the embedded relaxed problem are then proposed to obtain its global optimal solution: one is based on the branch-and-bound method and the other adds extra constraints in each iteration to update the iterative process when searching and approaching the global optimal solution. The main contribution of the algorithm is a tight lower-bound problem is formulated for the original problem via various linearization and convexification techniques without introducing any additional binary variables. Efficiency of the proposed algorithms are illustrated in the numerical studies.

Classical discrete network design problem assumes the capacities of the candidate links are given and only link addition is optimized. This thesis considers a novel discrete network design problem that aims to determine both the link addition decision (0 or 1) and the optimal link capacity (the optimal link capacity could be either continuous or at discrete levels) simultaneously, which was not addressed in previous studies or otherwise considered separately. A global optimization method employing linearization, outer approximation and range reduction techniques is developed to solve the formulated model. The computational results are consistent with the theoretical analysis, that is, the solution of the proposed model may provide a network design plan, which is generally better than plan from the sequential two-step method.

Finally, this thesis proposes a model to address an urban transport planning problem involving combined network design and signal setting in a saturated network. The mutual interaction between network capacity expansion and signal setting is considered so that best transport network performance can be guaranteed. A bi-level programming model is formulated and a global optimization solution method is designed based on mixed-integer linearization approach to solve the problem, which is inherently non-linear non-convex. Numerical example shows the proposed model provides a better result than ENSS and CNDP.

Chapter 1 INTRODUCTION

1.1 Background

The growth of urban population, social progress and economic prosperity lead to an increasing traffic demand. However, the capacity of most urban road network can hardly keep pace with even a small growth in traffic demand. This imbalance between urban traffic demand and supply can lead to frequent traffic congestion and thus incur significant negative effects to the social and economic developments.

In order to reduce traffic congestion in urban city area, transportation infrastructure system should be designed and reorganized to ensure that it has been utilized efficiently. Thus, the main focus in this thesis is the study of transportation network design problem (NDP), which is proposed to determine an effective urban transport planning with limited public resources to accommodate traffic demand at a fast growing rate. Traditionally, NDP handles optimal decision of road addition when constructing new road links to transportation system or determining the optimal capacity expansion of existing road network. It can also be generalized to handle road pricing problem in the network and signal control problem with traffic assignment. In addition, rather than being limited to road transportation systems, NDP can also be applied to other transportation modes, like public transport and shipping. Hence, this class of problems deserves more attention and further studies.

During the past few decades, NDP has always fascinated researchers because of the wide application in practice and its complication in theory. Review papers

on NDP includes Boyce (1984), Magnanti and Wong (1984), Friesz (1985), Migdalas (1995), Yang and Bell (1998) and Farahani et al. (2013). Typically, by making decisions on new road link construction or existing link expansion, NDP aims to optimize a specific performance index of the transportation system (e.g. minimize the total system travel cost) within government budget, while taking into account road users' routing behavior at the same time. It is a complicated operations research problem and is usually formulated as bi-level programming. The upper-level describes the traffic planners' decision-making process on transportation network expansion, which targets the optimal system performance index (e.g. investment and travel cost). The lower-level represents road users' routing behavior in response to the alteration of transportation network.

To solve an NDP is indeed to determine an optimal decision which costs less and achieves the best performance of transportation system. However, the mutual interaction between road users' route choice behavior and traffic planners' decision-making leads to the complex bi-level model formulation of NDP, which is usually nonlinear and nonconvex everywhere. Thus, even finding a local optimal solution of NDP is quite challenging. At first, researchers focus on studying local optimal solution algorithms, through which only local optimal points can be found, such as gradient based method, sensitivity analysis-base method and augmented Lagrangian method. Heuristic algorithms like genetic algorithm and simulated annealing algorithm are also applied to solve NDP. Though these algorithms are widely studied, global optimal solution of NDP still cannot be guaranteed, which means the solution obtained from the applied algorithm is not the true optimal transportation plan.

Noticing this problem, recently, Wang and Lo (2010) began to study the global optimal solution method for a conventional continuous network design problem. In their research, the nonconvex nonlinear model is firstly transformed into a mixed-integer linear programming (MILP) via piecewise-linear approximation of nonlinear function, then easily solved by existing MILP solver. Then Luatkep et al. (2011) extended the global optimal algorithm to a discrete network design problem and a mixed network design problem. Other researchers also began to study the global optimal solution algorithm in network design area soon afterwards (Ekström et al., 2012; Li et al., 2012; Wang et al., 2013). However, network design problem is a large research area as mentioned above and only a few problems have been addressed so far. Many more network design problems still need to be considered and their related global optimal solution algorithm should be explored. Secondly, it should be noted that in the study of Wang and Lo (2010) and Luatkep et al. (2011), the model formulation relies on piecewise linear approximation of the link travel time function, thus only an approximate global optimal solution other than exact global optimal solution of the original problem can be obtained. Hence, it is important to study how to improve the solution accuracy of the global optimal algorithm. Finally, the computational load of their methods is heavy which may require a long solution time for large size network, especially for real network. Considering the practical application in the future, efficiency enhancement of global optimal solution method also needs to be studied.

1.2 Objectives of the research

The objectives of the research are mainly including two parts: firstly, to develop more realistic mathematical programming models for urban transportation

network design problems; secondly, to propose efficient global optimization solution algorithms for network design problems and ensure the true globally optimal transportation planning.

The specific objectives of the research are:

- I To provide a global optimization method for the continuous network design problem with stochastic user equilibrium traffic assignment constraint and to guarantee the exact global optimal solution of the problem;
- II To present efficient global optimal solution methods for the discrete network design problem with stochastic user equilibrium constraint and given budget;
- III To develop a mathematical model describing a novel discrete network design 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. To propose a global optimal algorithm for the developed problem.
- IV To develop a mathematical model for the combined network design and signal setting problem in a saturated network and to propose a global optimal solution algorithm based on mixed-integer linearization approach to solve the developed model.

1.3 Outline of this thesis

The thesis consists of seven chapters, the organization of which is shown in Figure 1.1. Chapter 1 briefly introduces the background and motivations of this study. Chapter 2 reviews the literature on urban network design problems and summarizes the solution algorithm development of this type of problems.

The following four chapters are the main body of the thesis. Chapter 3 and chapter 4 separately address two traditional network design problems and study their corresponding global optimal solution methods. Chapter 5 and chapter 6 focus on improvement of the network design modeling and separately consider two unexplored urban transport network design problems and their global optimal algorithms.

Chapter 3 considers the continuous road network design problem with stochastic user equilibrium constraint that aims to optimize the network performance via road capacity expansion. In the literature, the classical continuous network design problem, widely believed as the most challenging frontier of transportation study, has received extensive research attentions. However, researchers have only recently begun to develop solution method to find the global optimization to CNDP, which is inherently formulated into a non-convex non-linear programming (the details are shown in the literature review). Nevertheless, most of the researches are focused on the CNDP with deterministic user equilibrium traffic assignment, while no previous research has ever tried to propose a global optimization solution approach for CNDP with stochastic user equilibrium (SUE). It should be noted that SUE is a more general user equilibrium principle, which is more realistic in describing the route choice behavior and most importantly include deterministic user

equilibrium (DUE) as a particular case. Hence, this chapter aims to fill this research gap that no global optimization solution method is yet to be developed for CNDP with SUE. A global optimization solution algorithm for CNDP with SUE is developed based on a tight linear programming relaxation. It guarantees exact global optimal solution property of the original CNDP with SUE, rather than only optimum of the linearized approximation of the original CNDP. Besides, the algorithm ascertains high solution efficiency by using range reduction technique.

Figure 1.1 Outline of the thesis.

Chapter 4 addresses the discrete road network design problem with stochastic user equilibrium constraint and given budget that aims to optimize the network performance via new link construction. Similarly, though it is more general and realistic than the DNDP with DUE problem, this problem, i.e., the DNDP with SUE, is rarely considered and no global optimal solution method is studied in the literature. This chapter aims to fill this research gap and develops two global optimal solution algorithms for the problem. The original problem is relaxed to a convex mixed-integer nonlinear problem, whose solution provides a lower bound of the original problem. The relaxed problem is then embedded into the two proposed global optimization solution algorithms: one is based on the branch-and-bound method and the other adds extra constraints in each iteration to update the iterative process when searching and approaching the global optimal solution.

Chapter 5 addresses a novel discrete network design problem that aims to determine the optimal new link addition and their optimal capacities simultaneously. Typically, the NDP problems are classified as either discrete network design problem, optimizing the plan of the new links addition; or the continuous network design problem, seeking the optimal road capacity expansion to existing network. However, in the classical DNDP problem model formulation, only link addition is considered and optimized, while the capacities of the candidate links are treated as given. The reason to assume the given and fixed new link capacity may be to simplify the model formulation so that appropriate solution algorithm could be developed to solve the difficult NP-hard problem. Indeed, to plan a new link addition to the existing network, both the link addition decision (0 or 1) and the optimal link capacity (the optimal link capacity could be either continuous or at discrete levels) should be considered simultaneously. This chapter fills this research gap in the literature

by formulating the network design problem in consideration of both the link addition and the link capacity optimization of the added links. By applying some linearization and relaxation techniques, a global optimization solution algorithm is proposed to solve the problem, and the global optimality is proved to be guaranteed.

Chapter 6 proposes an urban transport planning problem involving combined network design and signal setting in a saturated network. Basically, most of the previous research works investigated the two planning problems, road capacity design in NDP and network signal setting problem (ENSS), separately. However, in practice, NDP determines the link capacity, while ENSS determines the capacity allocation; and it is indeed the combination of the road capacity design and the capacity allocation design that fully determines the resultant equilibrium travel pattern and the transportation network performance. Previous studies addressed the two problems separately, regarding the other problem as exogenously given, which ignores the mutual interaction between them in determining the transport network performance. In this study, the two issues will be unified into one modeling framework as a combined network design and signal setting problem to optimize the transportation system performance. Moreover, a global optimization solution method based on mixed-integer linearization approach is designed to solve the formulated problem, which is a nonlinear and non-convex model.

Finally, chapter 7 is the conclusion and discussion part of the thesis. The findings and contributions of the thesis are concluded in this chapter, where a number of recommendations for future work are also provided.

Chapter 2 LITERATURE REVIEW

In the literature, network design problem has received extensive research attention for decades. Historically, the NDP has emerged to cater for the faster growing travel demand via the optimal decision on the expansion of urban road network, subject to the limited resources available for system improvement. The objective of the NDP is to optimize a specific performance index of the transportation system (e.g. minimize the total monetary travel cost), accounting for drivers' routing behavior or traffic assignment at the same time. The traffic assignment is usually in a manner of deterministic user equilibrium, which follows the Wardrop's first principle (Wardrop, 1952), or in a manner of stochastic user equilibrium (Sheffi, 1985), which assume the drivers may not perfectly know the road information. There are mainly three types of the NDP: one is dealing with the continuous capacity expansion of links, referred as continuous network design problem (Abdulaal and LeBlanc, 1979; Suwansirikul et al., 1987; Tan et al., 1979), another one determines the optimal decision of the new links addition, referred as discrete network design problem (Gao et al., 2005; Poorzahedy and Turnquist, 1982; Solanki et al., 1998), and a combination of the former two types, referred as mixed network design problem (MNDP) (Luathep et al., 2011).

This chapter reviews the literature on transport network design problems. Subsection 2.1 focuses on the CNDP models, including equilibrium signal setting problem, and development of the relevant solution algorithms. Subsection 2.2 briefly reviews the DNDP and their related solution methods. Subsection 2.3 provides a summary of the existing algorithms for network design problems.

2.1 Continuous network design problem

The continuous network design problem deals with determining the link capacity expansions to achieve a certain optimal objective, subject to the resultant flow pattern following some sort of equilibrium conditions.

Since the decision variables are continuous in CNDP, some traffic planning problems, which also deal with continuous decision variables, can be treated as a group of generalized CNDP, including road pricing problem and signal setting problem (Ferrari, 2002; Gori and Cipriani, 2000; Meneguzzo, 1997; Morrison, 1986; Yang and Bell, 1998; Yang and Huang, 2005). In the literature, the equilibrium network signal setting problem (ENSS) involves the optimal signal control of the transport system to improve the network performance (e.g. minimize delay, fuel consumption, or travel time). Studies on ENSS include two categories: local optimization of signal setting for individual intersection and global signal setting for the network. Local methods are widely studied in previous studies (Allsop, 1974; Cantarella et al., 2006; Gartner et al., 1980; Van Vuren and Van Vliet, 1992), which assume fixed and exogenously given network flows. On the other hand, global methods (Heydecker and Khoo, 1990; Marcotte, 1983; Sheffi and Powell, 1983; Yang and Yagar, 1995) determine signal setting parameters and endogenous equilibrium network flows simultaneously.

The solution of both CNDP and the ENSS problems is a challenging issue because the model formulations are intrinsically non-linear and non-convex, which makes it difficult even to find a solution algorithm for local optimum (Luo et al., 1996; Teklu et al., 2007). Sheffi and Powell (1983) proposed a gradient projection method on solving the ENSS problem and subsequently

extended in Cascetta et al. (1998), Lee and Machemehl (1998) and Cipriani and Fusco (2004). Chiou (2005) exploited another descent approach implementing four variants of gradient-based methods to solve the general CNDP. In addition, based on the results of Tobin and Friesz (1988), a number of sensitivity analysis based algorithms were introduced to solve bi-level transport planning and management problems: Friesz et al. (1990), Sumalee et al. (2006) and Connors et al. (2007) applied this method on the general CNDP and relevant problems; Yang and Yagar (1995) developed an algorithm to solve the ENSS with capacity constraints; Ying et al. (2007) further developed a sensitivity analysis based algorithm on ENSS, which significantly improved computing efficiency by avoiding the path-enumerating process. Moreover, Meng et al. (2001) imposed an augmented Lagrangian method to solve an equivalent single level problem of CNDP with non-convex and continuously differentiable properties, which was transformed via a marginal function tool from the bi-level program. However, for the non-convex non-linear CNDP and the ENSS problems, most solved from the Log-MILP method for the two scenarios with continuous and solution are heavily dependent on the selection of initial points. Besides, getting gradient information will be much more difficult with the increasing size of the network. The other type of solution method is metaheuristic algorithm. Abdulaal and LeBlanc (1979) suggested a Hooke-Jeeves algorithm to solve the CNDP by transferring the CNDP into an unconstrained non-linear program. Then a simple metaheuristic algorithm called equilibrium decomposed optimization method was developed by Suwansirikul et al. (1987) for practical use. It increased the computing efficiency through decomposing the CNDP into interacting sub-problems and calculating user equilibrium at each iteration. Besides, genetic algorithm (Ceylan and Bell, 2004; Lee and Machemehl, 1998; Lee and Machemehl, 2005; Teklu et al., 2007) and simulated annealing (Friesz

et al., 1993; Friesz et al., 1992) are also applied on the CNDP and ENSS problem. The advantage of the metaheuristic algorithms is that they are simple and can be easily implemented on the non-convex nonlinear problems with many local optimal solutions. What's more, as derivative information is not needed, they avoid the troublesome gradient calculation process (Lee and Machemehl, 2005). However, generally long computational time is needed in the stochastic global search, and the global optimal solution cannot be guaranteed.

Recently, Wang and Lo (2010) proposed a path-based algorithm seeking the global optimal solution of the CNDP by approximating the problem as a mixed-integer linear programming (MILP) problem. Li et al. (2012) developed a global optimization method based on the concepts of gap function and penalty for CNDP, which is then reformulated to a sequence of concave programming sub-problems. A multicutting plane method is used for solving each sub-problem.

2.2 Discrete network design problem

The discrete network design problem 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.

In the literature, Magnanti and Wong (1984) presented a unifying framework for deriving a bunch of algorithms for DNDP and reviewed some computational experience in solving NDP. A branch-and-bound based method is presented by Leblanc (1975) to solve the general DNDP. Solanki et al. (1998) proposed a

modified quasi-optimization heuristic algorithm by decomposing the original DNDP into a sequence of sub-problems. Gao et al. (2005) applied the support function concept to express the relationship between link addition plan and flows on the network, thus, the upper-level program of the DNDP could be reformulated as a general nonlinear programming and be easily solved by many traditional solution algorithms. In order to efficiently solve this problem, Poorzahedy and Turnquist (1982) developed a well-performed single-level model to approximate the original DNDP by suggesting a substitution for total user cost function. A B&B based heuristic algorithm was also given in their research.

Heuristic and meta-heuristic methods were also studied to solve DNDP. Poorzahedy and Abulghasemi (2005) implemented a cooperation agent or ant system method to solve DNDP. Kim and Kim (2006) suggested a genetic algorithm to solve their developed DNDP model. Drezner and Wesolowsky (2003) studied performances of several heuristic methods on different DNDP examples, including tabu search, simulated annealing and genetic algorithm. Some hybrid meta-heuristic methods were also been designed and tested on a number of example networks (Miandoabchi et al., 2012; Poorzahedy and Rouhani, 2007).

Global optimal solution methods for DNDP have recently drawn increasing attention from researchers in the transportation network design area. Wang and Lo (2010) approximated the bi-level CNDP model by a general mixed-integer linear programming and solved to its global optimum via traditional MILP solution algorithm. Luathep et al. (2011) adopted a variational inequality problem to describe DUE condition and proposed a link-based global optimal solution method for MNDP following the approach of Wang and Lo (2010) by

using MILP as approximation and taking advantage of cutting constraint technique. This MILP approach was extended to solve other transportation problems with equilibrium constraints (Ekström et al., 2013; Ekström et al., 2012; Zhang and van Wee, 2012). Farvaresh and Sepehri (2011) formulated the bi-level DNDP as a single-level MILP by casting the DUE constraints into the equivalent Karush-Kuhn-Tucker (KKT) condition. Wang et al. (2013) developed a DNDP with multiple capacity levels to decide how many lanes should be added to existing congested roads and designed a global approach based on the relationship of system optimal and user equilibrium traffic assignment. Furthermore, a benders decomposition method (Fontaine and Minner, 2014) was proposed for solving discrete-continuous linear bi-level problems in the context of transport network design.

2.3 Concluding remarks

In this chapter, models of NDP have been summarized, which are classified into CNDP, DNDP and MNDP according to different types of planning problems they are dealing with. Since this problem is often formulated as bi-level programming, which is difficult to solve even to obtain a local optimal solution, researchers have studied various solution methods for this family of problems. Development of solution methods for NDP is also stated in this chapter. More recently, researchers began to study global optimal solution algorithm for NDP, which can obtain the true optimal plan other than only a better plan of the problem. In the following chapters, several NDP problems will be addressed and the related global optimal solution algorithms will be provided.

Chapter 3 GLOBAL OPTIMIZATION METHOD FOR NETWORK DESIGN PROBLEM WITH STOCHASTIC USER EQUILIBRIUM

This chapter considers the continuous road network design problem with stochastic user equilibrium constraint that aims to optimize the network performance via road capacity expansion. The network flow pattern is subject to stochastic user equilibrium, specifically, the logit route choice model. The resulting formulation, a nonlinear nonconvex programming problem, is firstly transformed into a nonlinear program with only logarithmic functions as nonlinear terms, for which a tight linear programming relaxation is derived by using an outer-approximation technique. The linear programming relaxation is then embedded within a global optimization solution algorithm based on range reduction technique, and the proposed approach is proved to converge to a global optimum.

3.1 Introduction

There is a vast volume of literature published on the topic of continuous network design. Nevertheless, few studies in the literature dealt with the network design problem with stochastic user equilibrium travel pattern. Davis (1994) exploited standard nonlinear programming solution methods to find the exact local optimum of the network design problem with stochastic user equilibrium. Due to the inherent nonconvex property of the network design problem formulation, it is very hard to find the global optimal solution.

In this chapter, the continuous network design problem with SUE is considered, specifically, logit route choice model, and a global optimization algorithm to obtain the global optimum of the problem is proposed. The CNDP with SUE is investigated not only because it is more realistic in describing route choice behaviour, but also because SUE model is a more general statement of user equilibrium, which includes deterministic user equilibrium conditions as a special case. With the original nonlinear and nonconvex programming formulation, firstly logarithmic function of the nonlinear terms is taken, including multi-variable monomials describing the travel time/cost functions and the logit model. By doing so, the nonlinearity of the reformulated problem stems from constraints related to general concave logarithmic functions only. Then, an outer approximation technique is applied to construct a very tight linear programming relaxation. Specifically, by selecting break points to discretize the feasible domain of each logarithmic function, the associated tangential supports are used on the breakpoints, as well as chord lines connecting neighbouring break points, formulated by mixed-integer piecewise linearization, as the outer approximation of logarithmic function. Thus, a mixed-integer linear programming relaxation is derived, whose solution provides a lower bound of the original problem. The linear programming relaxation is then embedded within a global optimization algorithm based on a range reduction technique. The proposed approach is proved to converge to a global optimum.

The solution method proposed in this chapter is able to obtain the exact global optimum of the continuous network design problem with stochastic user equilibrium. The key lies in the nonlinearity transformation from the multivariable nonconvex functions in the original formulation to general single-variable concave logarithmic functions. By constructing an outer approximation

for the concave logarithmic function, a linear programming relaxation can be derived, whose solution is much easier to be obtained and thereby provide a lower bound of the original problem. It should be noted that the linear programming relaxation, formulated as a mixed-integer linear programming, is a very tight relaxation, and its solution will be sufficiently close to the global optimum if the discretization scheme applied is fine enough, which is exactly the spirit of global optimization solution methods to CNDP developed in some previous studies (e.g., Wang and Lo, 2010; Luatthep et al., 2011). However, the obtained solutions in these studies are only optimum of the linearized approximation of the original CNDP, rather than the exact solution to the original problem. In this chapter, the mixed-integer linear program is developed as a relaxation problem of the original model formulation. With the solution of the mixed-integer linear programming relaxation as the lower bound, a range reduction technique is used to propose an iterative solution algorithm, ensuring the exact global optimality of the solution to the original problem. Similar range reduction technique is also employed in Ekström et al. (2013) as a pre-processing scheme, which updated variable bounds only once before the branch-and-cut process and thus the MILP approximation of the original nonlinear problem still relies on large number of breakpoints. In this chapter, the range reduction will be embedded in an iterative process, thus the tightening of the relaxation model largely relies on the reduction of the variable bounds and rather smaller number of breakpoints is needed than the methods developed in the previous paper.

The proposed method in this chapter can also be applied to solve network design problems with DUE by either employing SUE model to approximate DUE via a large scaling parameter, or by adding additional binary variables and extra constraints to guarantee strictly positive value of variables in the

logarithmic functions. For illustration purpose, only continuous network design problem with SUE is elaborated in this chapter.

The remainder of this chapter is organized as follows: section 3.2 presents the model formulation and reformulation, section 3.3 describes the global optimization solution algorithm, and section 3.4 illustrates the computational study. Section 3.5 concludes the chapter with a summary.

3.2 Problem formulation

The following notation is used for the model formulation:

Parameters

A	Link set of the network
A^0	The set of links that are not candidates to be expanded
W	The set of origin-destination (O-D) pairs
R^w	The set of all existing paths between O-D pair $w \in W$
d^w	The travel demand between O-D pair $w \in W$
δ_{ar}^w	The link-path incidence, $\delta_{ar}^w=1$ if link $a \in A$ belongs to path $r \in R^w$ between O-D pair $w \in W$, otherwise $\delta_{ar}^w=0$
y_a	Current link capacity of link $a \in A$
$\bar{y}_a$	Maximum link capacity for link $a \in A$
T_a	Free flow travel time for link $a \in A$
R_a	The rate at which congestion increases the travel cost
ρ	The conversion factor which turns monetary cost into travel cost
θ	A positive parameter which scales the perceived travel cost

Variables

F	The objective function which denotes the total system cost
x_a	The traffic flow of link $a \in A$
y_a	The designed link a 's capacity after expansion for $a \in A$
g_a	Construction cost for expanding link $a \in A$, which is usually assumed to be a function of y_a
c_a	Travel time when using link $a \in A$, which is a function of both link flow and link capacity. $\mathbf{c} = [c_a]$, $a \in A$ denote the vector of all link travel times.
c_r^w	Time of traversing path $r \in R^w$ between O-D pair $w \in W$
p_r^w	The probability of path $r \in R^w$ is selected, which can be given as explicit functions of link travel costs
q_a^w	The probability of link $a \in A$ is chosen for a trip between O-D pair $w \in W$

3.2.1 Model formulation

The continuous network design problem with SUE flow pattern can be formulated as below:

$$\text{Minimize } F = \sum_{a \in A} x_a \cdot c_a(x_a, y_a) + \rho \sum_{a \in A} g_a(y_a) \quad (3.1)$$

Subject to

$$\underline{y}_a \leq y_a \leq \bar{y}_a, \quad a \in A \quad (3.2)$$

$$x_a = \sum_{w \in W} d^w \cdot q_a^w(\mathbf{c}), \quad a \in A \quad (3.3)$$

$$q_a^w(\mathbf{c}) = \sum_{r \in R^w} \delta_{ar}^w \cdot p_r^w(\mathbf{c}), \quad a \in A, \quad w \in W \quad (3.4)$$

$$p_{r_i}^w = \frac{e^{-\theta c_{r_i}^w}}{\sum_{r_j \in R^w} e^{-\theta c_{r_j}^w}}, \quad \forall r_i \in R^w, \quad \forall w \in W \quad (3.5)$$

$$c_r^w = \sum_{a \in A} \delta_{ar}^w \cdot c_a(x_a, y_a), \quad \forall r \in R^w, \quad \forall w \in W \quad (3.6)$$

In this formulation, the objective function seeks to minimize the combined cost for both network users, in terms of the total system travel time, and the associated capacity expansion cost. Constraint (3.2) imposes the road capacity expansion restriction. Constraints (3.3), (3.4), and (3.5) combine to enforce the stochastic user equilibrium flow pattern, wherein constraints (3.3) and (3.4) state the relationship between link flow and path flow, and constraint (3.5) entails the logit route choice model. Constraint (3.6) defines the total costs for using one specific path.

3.2.2 Model reformulation with logarithmic functions

3.2.2.1 Objective function

In this study, for illustration purposes, the typical BPR travel time function is adopted, expressed as:

$$c_a(x_a, y_a) = T_a \cdot \left[1 + R_a \cdot \left(\frac{x_a}{y_a} \right)^4 \right], \quad a \in A \quad (3.7)$$

where T_a and R_a are given parameters.

For the first term of the objective function,

$\sum_{a \in A} x_a \cdot c_a(x_a, y_a) = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a (x_a)^5 / (y_a)^4$, there is only one nonlinear part $(x_a)^5 / (y_a)^4$.

$$h_a = \frac{(x_a)^5}{(y_a)^4}, \quad a \in A \quad (3.8)$$

It should be noted, in equation (3.8), that all variables are strictly positive: firstly, link flow x_a cannot be zero under logit based SUE principle as the route choice probability is always positive due to the inherent property of logit function in (3.5); secondly, designed capacity of each link y_a is also strictly positive. Therefore, take the logarithm of this linear function on both sides of the equation and we have:

$$\ln(h_a) = 5\ln(x_a) - 4\ln(y_a), \quad \forall a \in A \quad (3.9)$$

In doing so, the monomials in the objective function is represented into equation (3.9), wherein the nonlinearity only lie in the logarithmic functions (i.e., other than the nonlinearity contained in the logarithmic functions themselves, equation (3.9) is indeed in linear form). Most importantly, the logarithmic function involved in equation (3.9) is single-variable function and globally concave, which facilitates the linear relaxation of the problem introduced in next section. For exposition purpose, if we let $l_{h,a} = \ln(h_a)$, $l_{x,a} = \ln(x_a)$, $l_{y,a} = \ln(y_a)$, equation (3.9) becomes:

$$l_{h,a} = 5l_{x,a} - 4l_{y,a} \quad \forall a \in A \quad (3.10)$$

The second term of the objective function, i.e. $\rho \sum_{a \in A} g_a(y_a)$, is used to calculate the total network expansion cost. The functional form of $g_a(y_a)$ may vary a lot, depending on the local geological and geographical conditions. In this study, for illustration purposes, $g_a(y_a)$ is considered to be linear function, as in previous studies (Meng et al., 2001; Wang and Lo, 2010). For links with

nonlinear cost functions, certain linearization method may be simply applied to approximate the nonlinear cost functions as linear constraints.

3.2.2.2 Link travel cost function

The link cost function (3.7) also appears in the constraints of CNDP. Following the same spirit as in the previous subsection, a new variable is used to represent the nonlinear part as below:

$$k_a = \left(\frac{x_a}{y_a} \right)^4, \quad a \in A \quad (3.11)$$

Therefore, the link cost function can be expressed as:

$$c_a(x_a, y_a) = T_a(1 + R_a k_a), \quad a \in A \quad (3.12)$$

As is done before, taking the logarithm of both sides of function (3.11) leads to

$$\ln(k_a) = 4\ln(x_a) - 4\ln(y_a), \quad a \in A \quad (3.13)$$

Similarly, letting $l_{k,a} = \ln(k_a)$, we have:

$$l_{k,a} = 4l_{x,a} - 4l_{y,a} \quad (3.14)$$

By doing so, the link travel cost function (3.7) can be expressed by linear equation (3.12) and linear logarithmic equation (3.14).

3.2.2.3 Logit route choice probability function

The logit model in (3.5) can be rewritten equivalently into:

$$\frac{p_{r_i}^w(\mathbf{c})}{p_{r_j}^w(\mathbf{c})} = \frac{e^{-\theta c_{r_i}^w}}{e^{-\theta c_{r_j}^w}}, \quad \forall r_i, r_j \in R^w, \quad r_i \neq r_j, \quad w \in W \quad (3.15)$$

$$\sum_{r \in R^w} p_r^w(\mathbf{c}) = 1, \quad w \in W \quad (3.16)$$

Taking the logarithm of both sides of equation (3.15), we have

$$\ln[p_{r_i}^w(\mathbf{c})] - \ln[p_{r_j}^w(\mathbf{c})] = \theta(c_{r_j}^w - c_{r_i}^w), \quad \forall r_i, r_j \in R^w, \quad r_i \neq r_j, \quad w \in W \quad (3.17)$$

To further simplify equation (3.17), let $l_{p,r_i} = \ln(p_{r_i}^w)$:

$$l_{p,r_i} - l_{p,r_j} = \theta(c_{r_j}^w - c_{r_i}^w), \quad \forall r_i, r_j \in R^w, \quad r_i \neq r_j, \quad w \in W \quad (3.18)$$

3.2.2.4 Model reformulation

In summary, the original CNDP is now cast into a reformulated model as follows, wherein the nonlinearity only lies in the logarithmic functions:

Reformulated Continuous Network Design Problem (RCNDP)

$$\text{Minimize } F = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a h_a + \rho \sum_{a \in A} g_a(y_a) \quad (3.19)$$

subject to

$$\underline{y}_a \leq y_a \leq \bar{y}_a, \quad a \in A \quad (3.20)$$

$$x_a = \sum_{w \in W} d^w \cdot q_a^w(\mathbf{c}), \quad a \in A \quad (3.21)$$

$$q_a^w(\mathbf{c}) = \sum_{r \in R^w} \delta_{ar}^w \cdot p_r^w(\mathbf{c}), \quad a \in A, \quad w \in W \quad (3.22)$$

$$l_{p,r_i} - l_{p,r_j} = \theta(c_{r_j}^w - c_{r_i}^w), \quad \forall r_i, r_j \in R^w, \quad r_i \neq r_j, \quad w \in W \quad (3.23)$$

$$\sum_{r \in R^w} p_r^w(\mathbf{c}) = 1, \quad w \in W \quad (3.24)$$

$$l_{p,r_i} = \ln(p_{r_i}^w), \quad \forall r_i \in R^w, \quad w \in W \quad (3.25)$$

$$c_r^w = \sum_{a \in A} \delta_{ar}^w \cdot c_a(x_a, y_a), \quad r \in R^w, \quad w \in W \quad (3.26)$$

$$l_{h,a} = 5l_{x,a} - 4l_{y,a} \quad \forall a \in A \quad (3.27)$$

$$l_{h,a} = \ln(h_a), \quad \forall a \in A \quad (3.28)$$

$$l_{x,a} = \ln(x_a), \quad \forall a \in A \quad (3.29)$$

$$l_{y,a} = \ln(y_a), \quad \forall a \in A \quad (3.30)$$

$$c_a(x_a, y_a) = T_a(1 + R_a k_a), \quad a \in A \quad (3.31)$$

$$l_{k,a} = 4l_{x,a} - 4l_{y,a}, \quad a \in A \quad (3.32)$$

$$l_{k,a} = \ln(k_a), \quad \forall a \in A \quad (3.33)$$

In this model reformulation, the nonlinearity stems only from the nonlinear constraints (3.25), (3.28), (3.29), (3.30) and (3.33).

3.3 Solution Algorithm

3.3.1 Outer approximation based linear programming relaxation

Noting the fact that the logarithmic functions in constraints (3.25), (3.28), (3.29), (3.30) and (3.33) are the only nonlinear terms in model RCNDP, a linear outer-approximation of these nonlinear constraints will be constructed to derive a linear programming relaxation of the reformulated model RCNDP. Without loss of generality, the nonlinear equation (3.30), $l_{y,a} = \ln(y_a)$, is taken as an example to elaborate on the linear relaxation method.

The feasible domain of y_a is bounded with two endpoints, $\underline{y}_a$ and $\bar{y}_a$. Within the interval $[\underline{y}_a, \bar{y}_a]$, a number of breakpoints are selected, denoted as y_a^i , as schematically shown in Figure 3.1. The outer-approximation of logarithmic curve $\ln(y_a)$ within feasible domain of $[\underline{y}_a, \bar{y}_a]$ is constructed via the tangential lines on each break point and endpoint and the chord lines connecting each pair of adjacent break points and endpoints, as shown in Figure 3.1, where only two break points are demonstrated for illustration purpose.

Figure 3.1 Linear relaxation for logarithmic function.

Specifically, the resulting linear outer approximation of logarithmic function in the two-dimensional space is given as below:

$$y_a^i = \underline{y}_a + \frac{\bar{y}_a - \underline{y}_a}{n-1} \cdot (i-1), \quad i = 1, 2, \dots, n \quad (3.34)$$

$$l_{y,a} \leq \ln(y_a^i) - 1 + \frac{y_a}{y_a^i}, \quad i = 1, 2, \dots, n \quad (3.35)$$

$$l_{y,a} \geq \underline{l}_{y,a} \quad (3.36)$$

$$y_a^i - M \cdot A_i \leq y_a \leq y_a^{i+1} + M \cdot A_i, \quad i = 1, 2, \dots, n-1 \quad (3.37)$$

$$\begin{aligned} \ln y_a^i + \frac{\ln y_a^{i+1} - \ln y_a^i}{y_a^{i+1} - y_a^i} \cdot (y_a - y_a^i) - M \cdot A_i &\leq \underline{l}_{y,a} \\ &\leq \ln y_a^i + \frac{\ln y_a^{i+1} - \ln y_a^i}{y_a^{i+1} - y_a^i} \cdot (y_a - y_a^i) + M \cdot A_i, \quad i = 1, 2, \dots, n-1 \end{aligned} \quad (3.38)$$

$$A_i = 0.5 \sum_{k=1}^{\zeta} (1 - (-1)^{\lfloor (i-1)/2^{k-1} \rfloor}) + \sum_{k=1}^{\zeta} ((-1)^{\lfloor (i-1)/2^{k-1} \rfloor} \cdot u_k), \quad i = 1, 2, \dots, n-1 \quad (3.39)$$

$$j = 1 + u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta \quad (3.40)$$

$$1 \leq j \leq n-1 \quad (3.41)$$

$$\zeta = \lceil \log_2(n-1) \rceil \quad (3.42)$$

$$u_k \in \{0,1\}, \quad k = 1, 2, \dots, \zeta \quad (3.43)$$

where y_a^i refers to the i th break point; n is the total number of break points including the two endpoints.

The right-hand side of inequality (3.35) represent all the tangential lines on each break point and endpoint, which constructs the upper bound of $\ln(y_a)$.

The chord lines connecting adjacent break points and endpoints will provide the lower bound of $\ln(y_a)$, denoted as $\underline{l}_{y,a}$ in constraint (3.36). The piecewise linear function is formulated into a mixed-integer linear constraints as described in (3.37) - (3.43), where $\lfloor x \rfloor$ refers to the integer part of x . As the formulation introduces binary variables, the number of binary variables will affect the computation efficiency significantly. This study follows the method proposed in Li and Lu (2009), which requires only the amount of $\lceil \log_2(n-1) \rceil$ binary variables if totally n breakpoints are needed, where $\lceil x \rceil$ means the smallest integer larger than or equal to value x . The choice of mixed-integer linear model affects the efficiency of the solution algorithm. Other existing model formulation approaches of piecewise-linear function (e.g. Log and DLog methods in Vielma et al. (2010)) can be used to replace the method illustrated in this chapter, up to the requirements of the solution efficiency in specific studies.

In equation (3.40), j refers to the index of chosen interval for the piecewise linear function. The feasible region of j is given in (3.41). Eq. (3.40) entails that each index number can be represented by a sum of powers of two, meanwhile, u_k is a binary variable and ζ is the maximum number of binary variables that is needed to express each index, which is stated in (3.43) and (3.42), respectively. For instance, assume $j = 7$ and $n = 9$, then j is equal to $1 + u_1 + 2u_2 + 2^2u_3$ if and only if $u_1 = 0$ and $u_2 = u_3 = 1$. The detailed proposition and proof describing the above remarks are shown as follows:

Proposition 1. For each index number $j \in [1, n-1]$, there is one unique group of binary numbers $u_k \in \{0, 1\}$, $k = 1, 2, \dots, \zeta$ that ensures the equation $j = 1 + u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta$, where $\zeta = \lceil \log_2(n-1) \rceil$ is the maximum number of binary numbers needed.

Proof. Since u_k is a 0-1 variable, let $(u_\zeta u_{\zeta-1} \dots u_k \dots u_1)$ stand for a binary number with the length of ζ bits. Thus, obviously, the binary variable $(u_\zeta u_{\zeta-1} \dots u_k \dots u_1)$ can equal to any decimal integer number between 0 to $1 + 2 + \dots + 2^{k-1} + \dots + 2^{\zeta-1} = 2^\zeta - 1$. Due to the binary variable $(u_\zeta u_{\zeta-1} \dots u_k \dots u_1)$ can also be written as $u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta$ in the decimal system, we have, $u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta \in [0, 2^\zeta - 1]$. Then, it can be deduced that $1 + u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta \in [1, 2^\zeta]$. That is to say, $1 + u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta$ can represent any integer number within the range $[1, 2^\zeta]$. Hence, in order to express $j \in [1, n-1]$, we need $n-1 \leq 2^\zeta$, i.e., $\zeta = \lceil \log_2(n-1) \rceil$. The proposition is proved.

In constraint (3.39), A_i is utilized to determine whether an interval on the feasible region of y_a is chosen and active: suppose variable y_a falls into the j th discrete interval, then $A_i = 0$ if $i = j$ and $A_i \geq 1$ if $i \neq j$. Let M in eq. (3.38) and (3.37) be a sufficiently large positive value, Eq. (3.37) demonstrates the bounds of y_a in selected interval (where $A_i = 0$) and Eq. (3.38) calculates the value of curve chord function of y_a , which provides the lower bound for the logarithmic function $\ln(y_a)$. Then the following proposition can be deduced.

Proposition 2. Given A_i in equation (3.39), $y_a \in [y_a^j, y_a^{j+1}]$, $j \in \{1, 2, \dots, n-1\}$ if and only if $A_i = 0$, $i = j$.

Proof. From the Proposition 1, it has already been proved that any index can be expressed by the formula $i = 1 + u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta$. Thus,

$$\begin{aligned}
A'_i &= 0.5 \sum_{k=1}^{\zeta} (1 - (-1)^{\lfloor (i-1)/2^{k-1} \rfloor}) \\
&= 0.5(\zeta - (-1)^{\lfloor (i-1)/2^0 \rfloor} - (-1)^{\lfloor (i-1)/2 \rfloor} - \dots - (-1)^{\lfloor (i-1)/2^{\zeta-1} \rfloor}) \\
&= 0.5(\zeta - (-1)^{u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta} - (-1)^{u_2 + 2u_3 + \dots + 2^{\zeta-2}u_\zeta} - \dots - (-1)^{u_\zeta}) \\
&= 0.5(\zeta - (-1)^{u_1} - (-1)^{u_2} - \dots - (-1)^{u_\zeta}) \\
&= 0.5 \sum_{k=1}^{\zeta} (1 - (-1)^{u_k})
\end{aligned} \tag{3.44}$$

$$\begin{aligned}
A''_i &= \sum_{k=1}^{\zeta} ((-1)^{\lfloor (i-1)/2^{k-1} \rfloor} \cdot u_k) \\
&= (-1)^{u_1} \cdot u_1 + (-1)^{u_2} \cdot u_2 + \dots + (-1)^{u_\zeta} \cdot u_\zeta \\
&= \sum_{k=1}^{\zeta} (-1)^{u_k} \cdot u_k
\end{aligned} \tag{3.45}$$

Hence, we have,

$$\begin{aligned}
A_i &= A'_i + A''_i \\
&= \sum_{k=1}^{\zeta} 0.5 - 0.5(-1)^{u_k} + (-1)^{u_k} \cdot u_k \\
&= \text{num}(B_1(i)) - \sum_{k \in B_1(i)} u_k + \sum_{k \in B_0(i)} u_k
\end{aligned} \tag{3.46}$$

where $B_0(i)$ is a set of all indexes k where $u_k = 0$, that is, $B_0(i) = \{k \mid u_k = 0, k \in 1, 2, \dots, \zeta\}$; similarly, $B_1(i)$ is a set of all indexes k where $u_k = 1$, i.e., $B_1(i) = \{k \mid u_k = 1, k \in 1, 2, \dots, \zeta\}$; $\text{num}(B_1(i))$ stands for the total number of elements in set $B_1(i)$.

(i) If $A_j = 0$, we have $\text{num}(B_1(j)) - \sum_{k \in B_1(j)} u_k = 0$ and $\sum_{k \in B_0(j)} u_k = 0$. This leads to a unique group of binary values. Thus, for $i \neq j$, there must be at least one $u_k = 1$ where $k \in B_0(j)$ or at least one $u_k = 0$ where $k \in B_1(j)$. Both conditions will result in $A_i \geq 1$ for $i \neq j$. Substituting these into equations (3.37) and (3.38), and noting M is an extremely large positive value, we have:

$$\left. \begin{aligned}
&-\infty \leq y_a \leq +\infty \\
&-\infty \leq l_{y,a} \leq +\infty, \quad i \in 1, 2, \dots, n \text{ and } i \neq j \\
&y_a^i \leq y_a \leq y_a^{i+1} \\
&l_{y,a} = \ln y_a^i + \frac{\ln y_a^{i+1} - \ln y_a^i}{y_a^{i+1} - y_a^i} \cdot (y_a - y_a^i), \quad i = j
\end{aligned} \right\} \tag{3.47}$$

$$\Rightarrow \left\{ \begin{aligned}
&y_a^j \leq y_a \leq y_a^{j+1} \\
&l_{y,a} = \ln y_a^j + \frac{\ln y_a^{j+1} - \ln y_a^j}{y_a^{j+1} - y_a^j} \cdot (y_a - y_a^j)
\end{aligned} \right.$$

(ii) If $y_a \in [y_a^j, y_a^{j+1}]$, $j \in \{1, 2, \dots, n-1\}$, it means that the j th interval is selected and active. Due to $j = 1 + u_1 + 2u_2 + \dots + 2^{\zeta-1}u_\zeta$, a unique group of

binary values can be obtained. Then, $\text{num}(B_1(j)) = \sum_{k \in B_1(j)} u_k$ and

$$u_k = 0, \forall k \in B_0(j), \text{ which leads to } A_j = 0.$$

This proposition is thus proved.

In summary, the constraints (3.36)-(3.43) ensures that the lower bound of $\ln(y_a)$ is chord lines connecting adjacent break points, which is indeed a piecewise linear function. Specifically, $l_{y,a} = \ln y_a^j + \frac{\ln y_a^{j+1} - \ln y_a^j}{y_a^{j+1} - y_a^j} \cdot (y_a - y_a^j)$ if y_a falls into the j th interval $[y_a^j, y_a^{j+1}]$. It should be noted that it requires only a logarithmic number of binary variables in this mixed-integer formulation.

So far, by using an outer-approximation technique, the nonlinear equation $l_{y,a} = \ln(y_a)$ is relaxed into a group of mixed-integer linear constraints. As this method is applicable to general logarithmic function, each logarithmic function in the RCNDP model can be represented by mixed-integer linear constraints.

For exposition purpose, let V be the set of variables, which are involved in logarithmic functions, i.e., $V = \{x_a, y_a, h_a, k_a, p_r^w\}$, $\forall a \in A, r \in R^w, w \in W$. Let l_v denotes the logarithmic function value $\ln v$ for any $v \in V$. For example, if $v = y_a, a \in A$, l_v stands for $l_{y,a}$. In summary, the mixed-integer linear programming relaxation can be stated as following:

Relaxed Mixed Integer Linear Program (RMILP)

$$\text{Minimize } F_L = \sum_{a \in A} T_a x_a + T_a R_a h_a + \rho \sum_{a \in A} g_a(y_a) \quad (3.48)$$

Subject to

$$\underline{y}_a \leq y_a \leq \bar{y}_a, \quad a \in A \quad (3.49)$$

$$x_a = \sum_{w \in W} d^w \cdot q_a^w(\mathbf{c}), \quad a \in A \quad (3.50)$$

$$q_a^w(\mathbf{c}) = \sum_{r \in R^w} \delta_{ar}^w \cdot p_r^w(\mathbf{c}), \quad a \in A, \quad w \in W \quad (3.51)$$

$$l_{p,r_i} - l_{p,r_j} = \theta(c_{r_j}^w - c_{r_i}^w), \quad r_i, r_j \in R^w, \quad r_i \neq r_j, \quad w \in W \quad (3.52)$$

$$\sum_{r \in R^w} p_r^w(\mathbf{c}) = 1, \quad w \in W \quad (3.53)$$

$$c_r^w = \sum_{a \in A} \delta_{ar}^w \cdot c_a(x_a, y_a), \quad r \in R^w, \quad w \in W \quad (3.54)$$

$$l_{h,a} = 5l_{x,a} - 4l_{y,a}, \quad a \in A \quad (3.55)$$

$$c_a(x_a, y_a) = T_a + T_a R_a k_a, \quad a \in A \quad (3.56)$$

$$l_{k,a} = 4l_{x,a} - 4l_{y,a}, \quad a \in A \quad (3.57)$$

$$\left. \begin{aligned} v^i &= \underline{v} + \frac{\bar{v} - \underline{v}}{n-1} \cdot (i-1), \quad i=1, 2, \dots, n \\ l_v &\leq \ln v^i - 1 + \frac{v}{v^i}, \quad i=1, 2, \dots, n \\ l_v &\geq \underline{l}_v \\ v^i - M \cdot A_i &\leq v \leq v^{i+1} + M \cdot A_i, \quad i=1, 2, \dots, n-1 \\ \ln v^i + \frac{\ln v^{i+1} - \ln v^i}{v^{i+1} - v^i} \cdot (v - v^i) - M \cdot A_i &\leq \underline{l}_v \\ &\leq \ln v^i + \frac{\ln v^{i+1} - \ln v^i}{v^{i+1} - v^i} \cdot (v - v^i) + M \cdot A_i, \quad i=1, 2, \dots, n-1 \\ A_i &= 0.5 \sum_{k=1}^{\zeta} (1 - (-1)^{\lfloor (i-1)/2^{k-1} \rfloor}) + \sum_{k=1}^{\zeta} ((-1)^{\lfloor (i-1)/2^{k-1} \rfloor} \cdot u_k), \quad i=1, 2, \dots, n-1 \\ j &= 1 + u_1 + 2u_2 + \dots + 2^{\zeta-1} u_{\zeta} \\ 1 &\leq j \leq n-1 \\ \zeta &= \lceil \log_2(n-1) \rceil \\ u_k &\in \{0, 1\}, \quad k=1, 2, \dots, \zeta \\ v &\in V = \{x_a, y_a, h_a, k_a, p_r^w\}, \quad \forall a \in A, r \in R^w, w \in W \end{aligned} \right\} \quad (3.58)$$

The number of break points has a great effect on the solution efficiency and quality of the relaxed MILP model shown above. More break points drive the outer-approximation closer to the original logarithmic function and tighter linear relaxation. In principle, if the number of break points is sufficiently large, the MILP model solution can be very close to the original CNDP model, which is indeed similar to the global optimization method proposed in Wang and Lo (2010).

With the linear relaxation construction of RCNDP, it is straightforward to derive the following proposition:

Proposition 3. The solution of the relaxed model provides the lower bound on the solution of the original CNDP model. If the variable solutions satisfy all the logarithmic function constraints, then it is also optimal to the original CNDP problem with same objective value.

Proof. Obvious from the construction.

3.3.2 Global optimization algorithm for CNDP applying range reduction technique

In this section, a range reduction technique based global optimization method is proposed to solve the problem.

3.3.2.1 Range reduction technique

In relaxing the original CNDP, a number of break points are selected to discretize the logarithmic curve in the feasible region, and thus binary variables are introduced in the linear programming relaxation. In principle, sufficiently large number of break points will lead to a much tighter relaxation and better lower bound solution. However, large amount of binary variables will result in heavy computation load and overwhelmingly long computation time. In other

words, the precision of the relaxation or approximation can be increased at price of introducing more binary variables and more computational load. To this end, the range reduction technique is introduced to reduce the feasible region while making sure the global optimal solution remains in the updated range. In this way, the variable ranges on which global optimum will not occur shall be removed to save the computation efforts. By doing so, with even a small number of breakpoints to partition the new range, the obtained lower bound is tighter and better, because the relaxation becomes much closer to the real logarithmic curve, as is illustrated schematically in Figure 3.2.

Figure 3.2 Methods of reducing error of the linear relaxation.

Specifically, an optimization-based range reduction technique is introduced as follows:

For an arbitrary variable $v \in V = \{x_a, y_a, p_r^w, h_a, k_a\}$, $\forall a \in A, r \in R^w, w \in W$, the improved new bounds of variable v , i.e., $\bar{v}^{new}$ and $\underline{v}^{new}$, can be obtained through the following MIP model:

$$\underline{v}^{new} = \text{Minimize } v \quad (3.59)$$

subject to

$$\underline{v}^{old} \leq v \leq \bar{v}^{old}, \quad \forall v \in V \quad (3.60)$$

$$F_L \leq \bar{F}_L^* \quad (3.61)$$

All the other constraints of the MILP model except bound constraints. (3.62)

$$\bar{v}^{new} = \text{Maximize } v \quad (3.63)$$

subject to

$$\underline{v}^{old} \leq v \leq \bar{v}^{old}, \quad \forall v \in V \quad (3.64)$$

$$F_L \leq \bar{F}_L^* \quad (3.65)$$

All the other constraints of the MILP model except bound constraints. (3.66)

where $\underline{v}^{old}$ and $\bar{v}^{old}$ are the incumbent lower and upper bounds for variable $v \in V$ respectively, $\bar{F}_L^*$ is a known upper bound of the global optimal objective function value of the original CNDP model. Hereby, it is important to obtain an upper bound to the solution of original CNDP, which can be achieved through different methods. For example, with a given a feasible road capacity expansion plan y , the corresponding traffic flow pattern x can be calculated by solving the equation constraints in (3.4)-(3.6). Any solution algorithm designed for solving logit based stochastic user equilibrium, like Dial's algorithm, can be applied. The objective function value in (3.1) based on the given y and associated SUE solution x can provide an upper bound to the solution of the original CNDP. Alternatively, to get an even better upper bound, one can employ the local optimum solution method proposed in Davis (1994).

After all variable bounds are updated using the above MILP model, bounds for h_a and k_a can be further tightened, because h_a and k_a are both functions of x_a and y_a . The domains of h_a and k_a can also be calculated when new bounds of

x_a and y_a are obtained. For h_a , $\forall a \in A$, since $h_a = (x_a)^5 / (y_a)^4$, letting $\underline{h}_a^{new1}$ be the new lower bound directly calculated from the range reduction MIP model and $\overline{h}_a^{new1}$ be the new upper bound, the bound of h_a , $\forall a \in A$ can be further tightened by adding the following additional constraints:

$$\underline{h}_a^{new2} = \frac{(\underline{x}_a^{new})^5}{(\underline{y}_a^{new})^4} \quad (3.67)$$

$$\overline{h}_a^{new2} = \frac{(\overline{x}_a^{new})^5}{(\overline{y}_a^{new})^4} \quad (3.68)$$

$$\underline{h}_a^{new} = \max\{\underline{h}_a^{new1}, \underline{h}_a^{new2}\} \quad (3.69)$$

$$\overline{h}_a^{new} = \min\{\overline{h}_a^{new1}, \overline{h}_a^{new2}\} \quad (3.70)$$

Similarly, for k_a , $\forall a \in A$, since $k_a = (x_a / y_a)^4$, k_a can be further tightened by adding the constraints below:

$$\underline{k}_a^{new2} = \frac{(\underline{x}_a^{new})^4}{(\underline{y}_a^{new})^4} \quad (3.71)$$

$$\overline{k}_a^{new2} = \frac{(\overline{x}_a^{new})^4}{(\overline{y}_a^{new})^4} \quad (3.72)$$

$$\underline{k}_a^{new} = \max\{\underline{k}_a^{new1}, \underline{k}_a^{new2}\} \quad (3.73)$$

$$\overline{k}_a^{new} = \min\{\overline{k}_a^{new1}, \overline{k}_a^{new2}\} \quad (3.74)$$

Remarks: Since updating bounds of variables also impacts the calculation efficiency, it is very important to carefully select the set of variables that needs range reduction or bound update in each iteration. For small network, bound tightening can be implemented on all the variables in set V . While for median or big size network, only updating a part of variables in set V is recommended for higher calculation efficiency. For example, we can simply update bounds of

variables in $\{x_a, y_a, p_r^w\}$, $\forall a \in A, r \in R^w, w \in W$. Then, new bounds of h_a and k_a , $\forall a \in A$ can be determined by the functional relationship with x_a, y_a .

3.3.2.2 Steps of proposed global optimization algorithm for CNDP

In this subsection, the detailed solution algorithm steps are specified:

Step 1. Select initial upper bound. Use a feasible solution of the objective function of the CNDP model as the initial upper bound $\overline{F}_0^*$, or other local optimal solution method to obtain an initial upper bound.

Step 2. Initialization. Set a small number as the total number of break points used in the RMILP model. This study initializes $n = 3$. The original feasible domain of each variable is used as their initial bound. Set the iteration number $t = 1$.

Step 3. Solve the relaxation problem. Relax the original CNDP model with the incumbent bounds of variables to develop an RMILP model. Solve the model to its global optimal solution by applying commercial software or conventional mathematical programming methods. Let $\mathbf{y}_{L,t}^*$ denote the road capacity expansion plan and $F_{L,t}^*$ the optimal objective function value obtained from the solution of the relaxed MILP model.

Step 4. Update the upper and lower bounds of the objective function, and evaluate the error. Use $\mathbf{y}_{L,t}^*$ as input road capacity enhancement plan to solve the traditional traffic assignment problem under the SUE condition, and thus the objective function value F_t^* of the original CNDP model. Update the upper bound of the objective function value of the CNDP model via $\overline{F}_t^* = \min\{\overline{F}_{t-1}^*, F_t^*\}$. Update the lower bound of the objective function value of

the CNDP model via $\underline{F}_t^* = \max\{\underline{F}_{t-1}^*, F_{L,t}^*\}$. The approximated global optimal solution of the CNDP model is improved to $\mathbf{y}_t^*$, which is solution of the RMILP model that is associated with current value of $\overline{F}_t^*$. Evaluate the upper bound and lower bound difference $|\overline{F}_t^* - \underline{F}_t^*| / \underline{F}_t^*$, and if $|\overline{F}_t^* - \underline{F}_t^*| / \underline{F}_t^* \leq \varepsilon$ ($\varepsilon \geq 0$), go to step 7.

Step 5. Update variable bounds. Apply the range reduction technique introduced above to calculate new bounds of each variable $v_i \in V$. Remember that always use the latest bounds to calculate next upper or lower bound of variable. Update the lower bound and upper bound of the variable range if the new bounds are tighter.

Step 6. Update the number of break points. Calculate the rate of each variable $r_v = (\overline{v}^{new} - \underline{v}^{new}) / (\overline{v}^{old} - \underline{v}^{old})$. If all r_v are larger than a given rate r^* ($0 < r^* < 1$), select m variables with the largest intervals. Increase the number of break points for the selected m variables by following $n = n + \hat{n}$, where $\hat{n}$ is a given integer number. Set the iteration number $t = t + 1$, go to step 3.

Step 7. List the global optimal solution $\mathbf{y}_t^*$.

3.3.2.3 Convergence

Proposition 4. The proposed algorithm converges to the global optimal solution $\hat{\mathbf{y}}$ of the original CNDP model and both the upper and lower bounds of the global optimal objective value converge to the exact global optimal solution of the CNDP model, when the iteration number t approaches to infinity.

Proof. Let $\hat{\mathbf{y}}$ be the exact global optimal solution of the original CNDP on the road capacity expansion and $\hat{F}$ be the associated objective function value. According to the process of relaxing logarithmic function, the optimal objective function value of the relaxation linear programming problem $F_{L,t}^*$ always underestimates the exact value of $\hat{F}$, i.e., $F_{L,t}^* \leq \hat{F}$. Since $\underline{F}_t^* = \max\{F_{t-1}^*, F_{L,t}^*\}$ is the maximum of all known optimal objective value of the MILP model, it maintains the property that $\underline{F}_t^* \leq \hat{F}$. Because $\overline{F}_t^* = \min\{\overline{F}_{t-1}^*, F_t^*\}$ is the minimum of all known upper bounds, it always overestimates the global optimal objective function value, i.e., $\hat{F} \leq \overline{F}_t^*$. Noting the fact that $F(\mathbf{y}_t^*) = \overline{F}_t^*$, we have $\underline{F}_t^* \leq \hat{F} \leq F(\mathbf{y}_t^*) = \overline{F}_t^*$. The implementations of range reduction technique and the method of increasing the number of break points in the presented algorithm guarantee that $\underline{F}_t^*$ and $\overline{F}_t^*$ can always be updated. Therefore, we have a monotonically increasing sequence $\underline{F}_1^* \leq \dots \leq \underline{F}_t^* \leq \dots \leq \underline{F}_{T-1}^* \leq \underline{F}_T^*$ and a monotonically decreasing sequence $\overline{F}_1^* \geq \dots \geq \overline{F}_t^* \geq \dots \geq \overline{F}_{T-1}^* \geq \overline{F}_T^*$. Therefore, if the iteration number t approaches to infinity, then $\lim_{t \rightarrow \infty} \underline{F}_t^* = \hat{F}$ and $\lim_{t \rightarrow \infty} \overline{F}_t^* = \lim_{t \rightarrow \infty} F(\mathbf{y}_t^*) = \hat{F}$. The convergence of the presented algorithm is proved.

3.4 Computational results

To evaluate the solution results from the proposed global optimal algorithm, three numerical case studies are implemented in this chapter. The first two examples follow Davis (1994) and their transportation networks are shown in Figure 3.3 and Figure 3.4. One is a small network, with six nodes, seven links

and four O-D pairs. The other network is of bigger size, with ten nodes, 14 links and nine O-D pairs. Input data of the seven-link network and the 14-link test network are shown in Table 3.2 and Table 3.3 respectively, including O-D travel demands, link characteristics and other parameters. In principle, the same parameter values as those adopted in Davis (1994) are used in priority. However, the parameters ρ and θ were not specified in that paper. Hence, the two parameters are set with reasonable values, which may not be identical with what were used in the literature. The last example is the well-known Sioux-Falls network in Figure 3.5, which has 24 nodes and 76 links. This network is used to do two tests. The first test in the subsection 3.4.1 is to test the performance of the proposed model and solution algorithm. The OD travel demand and link free flow travel time applied in this numerical test are the same with those used in Leblanc (1975). The link capacity and other parameters are listed in Table B.1 and Table B.2 of Appendix B. The six bold lines marked in the network are candidate links to be expanded. The same assumptions with Leblanc (1975) are also imposed that the two arcs between two nodes will be upgraded simultaneously. In order to illustrate that the proposed model and solution algorithm can also be applied to approximate the CNDP with DUE traffic assignment, the second test in the subsection 3.4.3 is conducted following the numerical examples used in Suwansirikul et al. (1987), Meng et al. (2001), Chiou (2005) and Luathep et al. (2011) to solve a CNDP with DUE problem. The ten dashed lines in Figure 3.5 are candidate links to be expanded. In this example, the parameter θ in the logit model is set sufficiently large to approximate DUE assignment by using SUE model (Sheffi, 1985). The comparison of the solution results from the proposed CNDP with SUE model and those from other papers are demonstrated in the subsection 3.4.3.

Table 3.1 Comparison of the test network size to those in the literature.

Literature	Algorithm	Largest test network					Solution quality
		Name	Node	Link	Candidate link	OD pair	
<i>Local method</i>							
LeBlanc (1975)	Branch-and-bound algorithm (DNDP)	Sioux-Falls network	24	76	5	552	Local
Abdulaal and LeBlanc (1979)	Hooke-Jeeves algorithm	Sioux-Falls network	24	76	5	552	Local
Suwansirikul et al. (1987)	Equilibrium decomposed optimization	Sioux-Falls network	24	76	10	552	Local
Davis (1994)	Sequential quadratic programming	Sioux-Falls network	24	76	10	552	Local
Meng et al. (2001)	Augmented lagrangian algorithm	Sioux-Falls network	24	76	10	552	Local
<i>Global method</i>							
Wang and Lo (2010)	Path-based MILP	16-link network	6	16	16	2	Global for approximated model
Luathep et al. (2011)	Link-based MILP	Sioux-Falls network	24	76	10	552	Global for approximated model
Li et al. (2012)	Penalty with multicutting plane method	16-link network	6	16	16	2	Global
Wang et al. (2012)	SO-relaxation based method (DNDP)	Sioux-Falls network	24	76	10	552	Global
		All test networks					
<i>This study</i>	linear relaxation based global optimization	Name	Node	Link	Candidate link	OD pair	
		7-link network	6	7	7	4	Global
		14-link network	10	14	14	9	Global
		Sioux-Falls network-1	24	76	3	552	Global
		Sioux-Falls network-2	24	76	10	552	Global

Figure 3.3 Seven-link test network.

Table 3.2 Data for the seven-link test network.

Link parameters					O-D demands	
a	T_a	R_a	$\underline{y}_a$	β	w	d^w
1	1.00	0.1500	200	3	O1-D1	500
2	0.25	0.0375	300	2	O1-D2	400
3	0.25	0.0375	700	1	O2-D1	400
4	0.25	0.0375	300	2	O2-D2	600
5	0.25	0.0375	300	2	Other parameters	
6	0.25	0.0375	300	2	ρ	0.03
7	1.00	0.1500	200	3	θ	30

Figure 3.4 14-link test network.

Table 3.3 Data for the 14-link test network.

Link parameters					O-D demands	
a	T_a	R_a	$\underline{y}_a$	β	w	d^w
1	1.00	0.1500	200	4	O1-D1	500
2	0.25	0.0375	600	2	O1-D2	400
3	0.25	0.0375	500	3	O1-D3	350
4	0.25	0.0375	600	2	O2-D1	400
5	0.25	0.0375	500	3	O2-D2	700
6	0.25	0.0375	500	3	O2-D3	200
7	0.25	0.0375	500	3	O3-D1	200
8	0.25	0.0375	500	3	O3-D2	250
9	0.25	0.0375	600	2	O3-D3	1000
10	0.25	0.0375	700	1		
11	0.25	0.0375	500	3		
12	0.25	0.0375	600	2	other parameters	
13	0.25	0.0375	600	2	ρ	0.001
14	0.25	0.0375	200	4	θ	30

In order to provide the reader a general sense of magnitude of problems solved in this chapter, the size of large test networks solved in similar papers in the literature is summarized in Table 3.1.

All computational experiments are performed 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 free toolbox YALMIP-R20130124 (Löfberg, 2004) together with MATLAB R2012a is adopted to model the two cases. The commercial optimization solver CPLEX optimization studio 12.3 (IBM ILOG, 2009) is used as an external solver of YALMIP to solve all MIP problems to their global optimal solutions.

3.4.1 Solution quality

Figure 3.5 Sioux-Falls network.

For the small test network, all bounds of variables in set V are updated in each iteration. The optimality tolerance for range reduction MILP problems is set to be 10^{-9} . The feasible objective function value or upper bound is initialized as $\overline{F}_0^* = 4489$, which is total system travel cost when doing nothing to the original network. The parameter r^* is set to be 0.95, m equals to 9 and $\hat{n}$ equals to 2. Set the tolerance ε in the presented algorithm for the error between the lower and upper bound of the global optimal objective function to be 10^{-6} . Then, the solution results of the proposed algorithm are illustrated in Table 3.4. It takes

seven iterations to reach the final global optimal solution and the computational time for the test is 10.00 minutes. The value of variables y_a shows the new capacity of each link in road expansion plan. The error between the upper and lower bounds is only 0.000060% ($\approx [\overline{F_T^*} - \underline{F_T^*}] / \underline{F_T^*}$), which indicates a extremely high solution quality. In Table 3.4, it is also demonstrated that the comparison between the link flow solutions directly obtained from the proposed algorithm and the exact link flow value calculated via the logit model based on the capacity expansion solution y_a . One can find that maximum error occurs on link 1 and is only 0.000075% ($= [337.8951 - 337.8948] / 337.8948$). This numerical example exactly illustrates the high solution quality and solution efficiency of the proposed solution algorithm.

Table 3.4 Results of the seven-link network by the proposed global optimal algorithm.

Variable	Value	Variable	Solutions directly from the algorithm	Exact value based on Logit Model
y_1	228.5674	x_1	337.8951	337.8948
y_2	318.6094	x_2	562.1049	562.1052
y_3	726.4955	x_3	1181.8928	1181.8933
y_4	318.5732	x_4	562.1049	562.1052
y_5	327.4224	x_5	619.7879	619.7881
y_6	327.4782	x_6	619.7879	619.7881
y_7	239.1410	x_7	380.2121	380.2119
T	7			
$\underline{F_T^*}$	3891.0479			
$\overline{F_T^*}$	3891.0502			
Time (mins)	10.00			

To show the efficiency of the proposed algorithm, the author also try to solve this numerical case by applying larger number of breakpoints while the range

reduction technique is not applied, i.e., the similar method as proposed in Wang and Lo (2010) is employed, wherein only chord lines connecting each pair of adjacent break points and endpoints are used to approximate the original nonlinear logarithmic function. Several different discretization schemes are separately employed in the linearization process. Results of these tests are listed in Table 3.5. With increasing number of the break points, calculation time rises fast. When using a five-breakpoint discretization schemes for each variable, the calculation time is only 0.009 hour; while this time rapidly rises to 5.3 hours if the nine-breakpoint discretization scheme is adopted. Compared to this method, the proposed algorithm in this chapter with range reduction technique is quite efficient by only costing 10 minutes of computational time. In addition, the results obtained through the method of Wang and Lo (2010) is only an approximated global optimal solution. Furthermore, comparing with the solution from the proposed method with range reduction, these results are found rather crude. The gaps between the exact objective values from the method of Wang and Lo (2010) and the one from the proposed method are 7.6%, 8.8% and 7.9% respectively. Though a better solution can be obtained by using a fine discretization scheme, it costs much longer time than the proposed algorithm with range reduction.

Table 3.5 Results of the seven-link network solved by the method of Wang and Lo (2010) in different partition schemes.

Breakpoint No.	5	7	9
MILP objective values	7687.8270	5897.7686	4955.9177
Exact objective values	4188.6940	4233.0665	4198.6886
Gap (%)	83.5	39.3	18.0
Time (hours)	0.009	0.3	5.3

Table 3.6 Results of the 14-link network by the proposed global optimal algorithm.

Y	Value	X	Solutions directly from the algorithm	Exact value based on Logit Model
y_1	250.0000	x_1	222.1941	221.7690
y_2	744.8002	x_2	1027.8059	1028.2310
y_3	561.6233	x_3	721.1165	721.5762
y_4	692.6474	x_4	877.8059	878.2310
y_5	541.4058	x_5	591.9045	591.7101
y_6	568.7918	x_6	684.3524	684.3514
y_7	567.0543	x_7	708.0955	708.2899
y_8	552.6815	x_8	665.6476	665.6486
y_9	723.0628	x_9	898.5938	898.3649
y_{10}	798.5292	x_{10}	841.0417	841.0062
y_{11}	625.0000	x_{11}	905.6120	905.6006
y_{12}	679.7710	x_{12}	772.6659	772.8843
y_{13}	750.0000	x_{13}	1005.6120	1005.6006
y_{14}	250.0000	x_{14}	544.3880	544.3994
T	9			
$\underline{F}_T^*$	4478.2136			
$\overline{F}_T^*$	4479.7471			
Time (hours)		2.32		

For the 14-link test network, all bounds of variables in set V are updated in each iteration. The optimality tolerance for range reduction MILP problems is 10^{-8} . The feasible objective function value is initialized with $\overline{F}_0^* = 5645$, which is the system travel cost on the original network. The value of parameter r^* , m and $\hat{n}$ are identical with those used in the seven-link network. The tolerance ε in the proposed algorithm is set equal to 5×10^{-4} . The solutions of this numerical study by using the proposed algorithm are shown in Table 3.6. The error between the upper and lower bounds after nine iterations is 0.034% ($\approx$

[4479.7471-4478.2136]/4478.2136). Therefore, the approximated solution can be contended to be very close to the exact global optimal solution of the original CNDP model. This test costs about 2.32 hours to reach the optimal solution.

Since all variable bounds are updated in each iteration, to reveal how much the ranges are reduced, Table 3.7, Table 3.8 and Table 3.9 respectively provide the computational results in each iteration about the remaining portion of different variable range, including link flow, link capacity and path selection probability. The remaining feasible range ensures that the optimal solution of current RMILP is not excluded. In this numerical example, the feasible ranges of x_a and p_r^w shrink in almost every iteration, while the range of y_a remains unchanged until the sixth iteration. This may illustrate that before the sixth iteration, x_a and p_r^w have greater impact on the solution of RMILP than y_a ; however, y_a starts to contribute to the solution efficiency after the sixth iteration. Another key point noticed is that the average remaining ratio of feasible range for x_a , y_a and p_r^w is 4.2%, 48.8% and 15.1% respectively. The minimum remaining ratio of feasible range even reaches 0.3%. It shows that the size of the final relaxed problem is much smaller than the original one and the range reduction technique has a significant effect in this process.

The Sioux-Falls network is used to illustrate performance of the proposed algorithm on large-scale network. The total travel time on the original network, in which the traffic assignment follows SUE principle, is employed as the initial feasible objective function value, that is, $\overline{F}_0^* = 3.6436 \times 10^8$. The computational results of Sioux-Falls network are reported in Table 3.10. Within the given upper bound of candidate link capacity (Table B.2 in Appendix B), the optimal

Table 3.7 Remaining percentage of feasible range of link flows in each iteration.

Iteration	1	2	3	4	5	6	7	8	9
x_1	100.0%	98.6%	82.3%	67.8%	54.8%	43.5%	31.9%	17.9%	7.6%
x_2	100.0%	96.7%	76.9%	61.2%	48.7%	39.1%	28.2%	16.4%	6.6%
x_3	100.0%	87.0%	63.2%	50.4%	40.7%	28.0%	17.5%	10.3%	5.0%
x_4	100.0%	84.7%	74.4%	60.7%	48.2%	38.2%	27.4%	14.9%	5.7%
x_5	100.0%	97.7%	89.9%	73.6%	51.5%	32.5%	21.3%	13.2%	5.4%
x_6	100.0%	96.4%	87.6%	71.5%	48.4%	30.2%	19.6%	12.1%	4.9%
x_7	100.0%	97.1%	87.1%	68.1%	46.1%	29.5%	19.7%	11.9%	4.8%
x_8	100.0%	93.8%	84.0%	66.0%	44.3%	27.4%	18.1%	11.1%	4.5%
x_9	100.0%	100.0%	89.6%	73.8%	50.3%	31.4%	21.3%	12.1%	5.1%
x_{10}	100.0%	100.0%	97.0%	82.2%	55.8%	33.6%	21.6%	12.5%	5.1%
x_{11}	100.0%	84.6%	38.2%	19.4%	16.0%	14.1%	5.4%	1.9%	0.4%
x_{12}	100.0%	84.1%	53.7%	39.4%	29.4%	18.3%	11.5%	6.9%	2.7%
x_{13}	100.0%	81.5%	30.4%	18.1%	16.0%	11.9%	4.6%	1.8%	0.4%
x_{14}	100.0%	79.0%	25.5%	17.8%	16.0%	10.9%	4.2%	1.7%	0.4%

Table 3.8 Remaining percentage of feasible range of link capacities in each iteration.

Iteration	1	2	3	4	5	6	7	8	9
y_1	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	42.5%
y_2	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	75.2%	37.6%
y_3	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	89.1%
y_4	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	64.4%
y_5	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	66.6%
y_6	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	66.5%
y_7	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	64.4%
y_8	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	62.7%
y_9	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	94.6%	49.5%
y_{10}	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	69.7%
y_{11}	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	52.3%	23.8%	5.1%
y_{12}	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	55.6%
y_{13}	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	58.4%	30.6%	8.1%
y_{14}	100.0%	100.0%	100.0%	100.0%	100.0%	83.9%	31.1%	11.5%	1.8%

Table 3.9 Remaining percentage of feasible range of p_r^w in each iteration.

Iteration	1	2	3	4	5	6	7	8	9
p_1^{11}	100.0%	81.8%	67.2%	54.9%	44.0%	31.6%	18.6%	10.5%	4.1%
p_2^{11}	100.0%	81.5%	67.0%	54.6%	43.8%	31.5%	18.6%	10.5%	4.1%
p_1^{12}	100.0%	98.0%	96.9%	93.3%	75.4%	42.6%	23.8%	14.2%	5.6%
p_2^{12}	100.0%	98.0%	96.8%	93.0%	71.8%	39.8%	23.1%	14.0%	5.5%
p_1^{13}	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
p_1^{21}	100.0%	98.0%	97.0%	93.5%	72.1%	39.8%	23.0%	13.5%	5.3%
p_2^{21}	100.0%	98.0%	97.0%	93.3%	68.6%	36.7%	22.0%	13.2%	5.2%
p_1^{22}	100.0%	95.6%	72.9%	52.0%	35.6%	20.7%	12.4%	7.3%	2.9%
p_2^{22}	100.0%	96.0%	88.0%	68.1%	48.3%	25.0%	15.6%	9.7%	3.7%
p_3^{22}	100.0%	96.0%	87.2%	66.6%	45.3%	30.5%	17.7%	10.2%	3.9%
p_4^{22}	100.0%	96.0%	93.1%	79.2%	55.9%	28.8%	17.8%	10.5%	4.0%
p_1^{23}	100.0%	98.0%	97.4%	94.6%	78.0%	49.7%	29.9%	18.0%	6.9%
p_2^{23}	100.0%	98.0%	97.4%	94.4%	75.5%	45.1%	28.6%	17.5%	6.7%
p_1^{31}	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
p_1^{32}	100.0%	98.0%	97.6%	94.3%	72.3%	44.3%	27.2%	16.1%	6.3%
p_2^{32}	100.0%	98.0%	97.5%	94.2%	69.4%	40.3%	25.8%	15.7%	6.2%
p_1^{33}	100.0%	39.4%	20.6%	16.9%	15.5%	10.2%	4.0%	1.6%	0.3%
p_2^{33}	100.0%	39.4%	20.5%	16.9%	15.5%	10.2%	4.0%	1.6%	0.3%

construction plan is to improve capacity of links (6,8), (10,16) and (13,24) to 9.2416, 6.4856 and 5.8794, respectively. The total cost of this solution can be reduced by 9.99% ($\approx [3.6436\text{E}+08 - 3.2796\text{E}+08] / 3.6436\text{E}+08$) compared to the original situation. Since this network is large, the loop is stopped when the error between the upper bound and the lower bound of the objective function is 3.85% ($\approx [3.2796\text{E}+08 - 3.1557\text{E}+08] / 3.1557\text{E}+08$). The total calculation time for this test is about 14.5 hours because the binary variable number significantly increases with the size of network and thus the computational load is rather high to obtain the global optimal solution.

Table 3.10 Results of the Sioux-Falls network via the proposed optimal algorithm.

Link pair	Capacity	Variable	Exact value	Variable	Exact value	Variable	Exact value	Variable	Exact value
(6,8)	9.2416	x_1	135.3464	x_{21}	186.1033	x_{41}	162.1941	x_{61}	234.8327
(10,16)	6.4856	x_2	146.7332	x_{22}	70.5514	x_{42}	86.2546	x_{62}	78.9260
(13,24)	5.8794	x_3	120.7605	x_{23}	141.1230	x_{43}	87.6084	x_{63}	62.9043
		x_4	124.5278	x_{24}	70.1997	x_{44}	76.1925	x_{64}	69.0020
		x_5	187.2543	x_{25}	84.6134	x_{45}	221.6274	x_{65}	66.7854
		x_6	192.8057	x_{26}	155.5472	x_{46}	93.5026	x_{66}	75.2103
		x_7	124.1257	x_{27}	75.4545	x_{47}	157.1169	x_{67}	91.5315
		x_8	122.9445	x_{28}	79.9409	x_{48}	150.0584	x_{68}	154.5438
		x_9	115.6628	x_{29}	151.5774	x_{49}	100.9980	x_{69}	79.6250
		x_{10}	99.4295	x_{30}	69.5515	x_{50}	97.6544	x_{70}	74.4624
		x_{11}	82.9713	x_{31}	173.4542	x_{51}	190.3588	x_{71}	94.2254
		x_{12}	117.3842	x_{32}	221.8024	x_{52}	86.0100	x_{72}	72.4362
		x_{13}	80.2961	x_{33}	85.8484	x_{53}	94.1073	x_{73}	108.5860
		x_{14}	72.8748	x_{34}	85.2027	x_{54}	73.5003	x_{74}	71.1678
		x_{15}	120.0593	x_{35}	179.2001	x_{55}	151.7798	x_{75}	96.1605
		x_{16}	165.0641	x_{36}	78.7595	x_{56}	213.7818	x_{76}	81.7721
		x_{17}	161.0746	x_{37}	84.1165	x_{57}	216.1771		
$\underline{F_T^*}$	3.1557E+08	x_{18}	187.1656	x_{38}	141.6483	x_{58}	157.3508		
$\overline{F_T^*}$	3.2796E+08	x_{19}	161.5870	x_{39}	81.9525	x_{59}	70.1149		
Time (hours)	14.5	x_{20}	86.6176	x_{40}	177.4274	x_{60}	82.0775		

3.4.2 Effect of the initial feasible objective value

In this solution algorithm, the initial lower bound of the problem is determined by the discretization scheme, basically, how many break points are introduced; meanwhile, the selection of initial upper bound is more arbitrary. In this section, the impact of the initial feasible objective function value $\overline{F}_0^*$ on result is also studied so as to test whether the initial value of $\overline{F}_0^*$ will affect computational efficiency and solution quality significantly.

Figure 3.6 shows convergence process of bounds of global optimal objective function value when using different $\overline{F}_0^*$ value in the seven-link network. Here, three cases are tested: (1) $\overline{F}_0^*$ is set equal to 3950, which is very close to the global optimal objective value. (2) $\overline{F}_0^*$ is set equal to 4489, which is equal to total system travel cost when no improvement is made on the original network. So this case will yield the maximum iteration number. (3) $\overline{F}_0^*$ is set equal to 4200, which is a number between the above two cases to be a comparison. The results of convergence process of the 14-link network are depicted in Figure 3.7. Three cases are also tested for this network. The three scenarios are very similar with the test in the smaller network: (1) $\overline{F}_0^*$ is equal to 4600; (2) $\overline{F}_0^*$ is equal to 5645; (3) $\overline{F}_0^*$ is equal to 5100.

From Figure 3.6 and Figure 3.7, it can be found that all cases are convergent to the global optimal solution, no matter how crude the initial objective value upper bound is. The bounds of the global optimal objective function value obtained are [3891.0478, 3891.0501], [3891.0479, 3891.0502] and [3891.0483, 3891.0500], respectively, for the above three cases of the seven-link network,

which only have slight difference among each other. For the bigger network, the final bounds of the objective function value are [4478.3607, 4479.6758], [4478.2136, 4479.7471] and [4478.2134, 4479.7389] for each of the above situation. They are also very close to each other. This fact illustrates that the initial value of $\overline{F}_0^*$ has little effect on the final precision of the global optimal solution.

Figure 3.6 Update process of upper and lower bounds of the global optimal objective value with different initial feasible objective value on seven-link network.

Another interesting finding is that the same number of iterations is used to reach the final solutions within a given tolerance for different cases in each network. For the three cases of the seven-link network, the computational time is 10.03 minutes, 10.00 minutes and 8.91 minutes, respectively. They do not

have much difference from each other. Similarly, for the 14-link network, the computational time does not vary considerably. The computational times are, respectively, 2.17 hours, 2.32 hours and 2.65 hours for the three cases.

Figure 3.7 Update process of upper and lower bounds of the global optimal objective value when adopting different initial feasible objective value on 14-link network.

3.4.3 Approximate CNDP with DUE problem by adopting a large θ value

In this subsection, the proposed model of CNDP with SUE and its solution algorithm are used to approximately solve the CNDP with DUE conditions and to compare the obtained results with those in the literature. As DUE is a special case of SUE (Sheffi, 1985), in principle, the obtained global optimal solution of the proposed model is also the global optimal solution of CNDP with DUE

traffic assignment if the parameter θ is set to be sufficiently large. In this study, a very high value of the parameter θ is used to ensure that the solution from this CNDP model with SUE can provide a good approximation of CNDP with DUE. The CNDP with SUE model is firstly set with a sufficiently large parameter θ , and then solved by the proposed algorithm to its global optimum. Using the obtained link capacity solution as network input parameters and solving the traditional DUE traffic assignment, the corresponding real objective value of CNDP model with DUE can be derived, which can be used to test the solution performance as compared to the solutions from previous studies.

The test network is shown in Figure 3.5, the Sioux-Falls network, and ten candidate links are marked with dashed lines. The upper bounds of the candidate link capacities are given in Table B.3 of Appendix B and all the other input information are the same with those used in Suwansirikul et al. (1987). For convenience of comparison, the abbreviations of solution algorithms for solving CNDP with DUE problem in the literature are listed in Table 3.11.

Table 3.11 Abbreviations of solution algorithms and references for solving CNDP with DUE problem.

Abbreviation	Solution algorithm	Reference
IOA	Iterative optimization-assignment algorithm	Allsop (1974)
HJ	Hooke–Jeeves algorithm	Abdulaal and LeBlanc (1979)
EDO	Equilibrium decomposed optimization	Suwansirikul et al. (1987)
SA	Simulated annealing algorithm	Friesz et al. (1992)
SAB	Sensitivity analysis-based algorithm	Yang and Yagar (1995)
AL	Augmented Lagrangian algorithm	Meng et al. (2001)
GP	Gradient Projection method	Chiou (2005)
CG	Conjugate gradient projection method	Chiou (2005)
QNE	Quasi-Newton projection method	Chiou (2005)
PT	PARATAN version of gradient projection method	Chiou (2005)
LMILP	Link based mixed-integer linear program	Luathep et al. (2011)

Table 3.12 Model results for approximated CNDP with DUE.

Variable	IOA	HJ	EDO	SA	SAB	AL	GP	CG	QNEW	PT	LMILP	This study
Initial value of y	12.5	2	12.5	6.25	12.5	12.5	6.25	12.5	12.5	12.5	–	–
y_{16}	4.6875	4.8	4.59	5.38	5.7392	5.5728	5.4277	4.7691	5.3052	5.0237	5.362	5.2001
y_{17}	3.9063	1.2	1.52	2.26	5.7182	1.6343	5.3235	4.8605	5.0541	5.2158	2.057	1.8545
y_{19}	1.2695	4.8	5.45	5.5	4.9591	5.6228	1.6825	3.0706	2.4415	1.8298	5.486	5.2001
y_{20}	1.6599	0.8	2.33	2.01	4.9612	1.6443	1.6761	2.6836	2.5442	1.5747	1.895	1.8545
y_{25}	2.3331	2	1.27	2.64	5.5066	3.1437	2.8361	2.8397	3.9328	2.7947	2.556	3.0374
y_{26}	2.3438	2.6	2.33	2.47	5.5199	3.2837	2.7288	2.9754	4.0927	2.6639	2.618	3.0374
y_{29}	5.5651	4.8	0.41	4.54	5.8024	7.6519	5.7501	5.6823	4.3454	6.1879	3.741	3.7842
y_{39}	4.6862	4.4	4.59	4.45	5.5902	3.8035	4.9992	4.2726	5.2427	4.9624	4.551	4.5982
y_{48}	5.4688	4.8	2.71	4.21	5.8439	7.382	4.4308	4.4026	4.7686	4.0674	3.741	3.7842
y_{74}	6.25	4.4	2.71	4.67	5.8662	3.6935	4.3081	5.5183	4.0239	3.9199	4.489	4.5982
Total travel time cost	77.576	76.366	80.135	75.670	73.437	74.664	77.591	76.097	76.576	77.804	76.011	75.9709
Improvement cost	6.604	5.079	3.132	5.487	10.796	8.743	6.483	6.629	6.456	6.295	4.910	4.9550
F	84.180	81.445	83.267	81.157	84.233	83.407	84.074	82.726	83.032	84.099	80.921	80.9259
F_{ori}	87.34	81.25	83.47	80.87	84.21	81.752	82.57	82.53	83.07	82.57	80.883	–

Table 3.12 compares the results from the proposed method with the parameter setting of $\theta=3000$ and those results from solution algorithms specifically for solving CNDP with DUE model in the literature. In Table 3.12, F represents the recalculated objective function value by using the link capacity solution as input and then solving the traditional DUE traffic assignment problem. F_{ori} denotes the original values that were reported in the papers in the literature. The CNDP with SUE model is firstly solved by directly applying the proposed solution algorithm to its optimal objective value of 81.0093. Then, the obtained link capacity expansion is used as known parameters and the DUE traffic assignment is solved, and the resultant objective value F is 80.9259. The gap between the two objective values is only 0.10%, which demonstrates that the approximated DUE from the SUE is already very close to the real DUE traffic assignment. Next, comparing the results to those from the literature, one can find the presented method provides very good solution quality in solving CNDP with DUE, with the second best value of F , only next to the result from Luathep et al. (2011). This can be interpreted by the fact that Luathep et al. (2011) developed a global optimal solution method to solve this CNDP with DUE particularly, while the method in this study, designed for CNDP with SUE, can only solve the approximated problem by setting a large value of θ . However, even so, the result gap between them is quite small, only around 0.006% ($\approx [80.9259-80.921]/80.921$). In summary, this numerical test shows that the proposed model and solution method can also serve as a good approximation to solve CNDP with DUE conditions, up to a certain accuracy requirement.

3.5 Concluding remarks

In this chapter, the continuous network design problem with stochastic user equilibrium assignment is considered. A global optimization method is proposed for solving the problem. The solution method firstly reformulates the original problem to transform the nonlinearity form. Then, it applies an outer-approximation approach to derive a tight linear program relaxation. Based on range reduction technique, a global optimization algorithm is developed. Computational examples are implemented to demonstrate the solution quality and efficiency of the proposed solution algorithm.

The presented algorithm in this chapter can be regarded as a framework of a global optimization solution method. The algorithm can be updated by adopting more advanced mixed-integer models in the linearization process, or by adopting more efficient range reduction technique. The present algorithm can also be easily extended to solve other discrete network design problem and mixed network design problem, which in essence can be cast into nonlinear and nonconvex programming problems.

Chapter 4 MODELING AND SOLVING DISCRETE NETWORK DESIGN PROBLEM WITH STOCHASTIC USER EQUILIBRIUM

The preceding chapter considers a continuous network design problem with stochastic user equilibrium. In this chapter, the discrete network design problem is addressed, which determines the addition of new roads to an existing transportation network with given budget to optimize the total transportation system performance. Same as last chapter, road users are also assumed to follow the traffic assignment principle of stochastic user equilibrium, specifically, the logit route choice model. A mixed-integer nonlinear nonconvex problem is firstly developed to model this DNDP with SUE. In order to obtain the global optimal solution of this model, the original problem is relaxed to a convex mixed-integer nonlinear problem, whose solution provides a lower bound of the original problem. The relaxed problem is then embedded into two proposed global optimization solution algorithms: one is based on the branch-and-bound method and the other adds extra constraints in each iteration to update the iterative process when searching and approaching the global optimal solution. In both of the two methods, solving relaxed problem provides an updated lower bound of the objective function value, while solving the SUE traffic assignment with the updated solution of relaxed problem provides an upper bound. Numerical experiments are conducted to test and illustrate the performance of the developed model and solution algorithms.

4.1 Introduction

In defining NDP, one key issue is the assumption of the travelers' routing choice behavior principle. While most of the DNDP models in the literature applied DUE, this chapter addresses the DNDP with SUE, and most importantly, a global optimization solution method is developed to solve this problem. The DNDP with SUE is investigated not only because it is more realistic in describing route choice behavior, but also because SUE model is a more general statement of user equilibrium, which includes DUE conditions as a special case (Sheffi, 1985). Davis (1994) proposed two standard nonlinear programming approaches for CNDP with SUE, which cannot handle DNDP. Liu and Wang (2015) considered the network design problem with SUE; Wang et al. (2013) proposed global optimization solution methods for discrete network design problem, however, only deterministic user equilibrium principle was applied. Besides, both of these two studies only considered optimizing the discrete levels of road capacity for existing network, that is, they assumed fixed road network topology and optimized the existing links' capacity in discrete level, while optimal new links addition was not addressed. Wang et al. (2015) developed a novel discrete network design problem with deterministic user equilibrium, which cannot deal with SUE traffic assignment. In fact, very few previous research works considered DNDP with SUE due to its computational difficulty. Chen and Alfa (1991) adopted a stochastic incremental traffic assignment approach for the presented DNDP problem, which may achieve a traffic assignment close to SUE but cannot be guaranteed. Indeed, the solution method cannot ensure that a global optimal solution is obtained. Therefore, this chapter aims to fill in the research gap by developing global optimization solution method to solve the formulated discrete network design problem with

stochastic user equilibrium principle. Particularly, the logit model, one of most typical SUE models, is employed to describe the road users' routing choice behavior.

In this chapter, a mixed-integer nonlinear programming model is developed for the DNDP with SUE, which is inherently nonconvex and challenging to be solved even to its local optimum. In contrast to previous method (Chen and Alfa, 1991), global optimal solution methods are developed in this chapter, which can solve the network design problems to their global optima, i.e. the globally best plan for network construction to achieve certain objective. To be more specific, this chapter provides two tractable global optimal algorithms to solve the presented problem of DNDP with SUE. To obtain the global optimal solution, firstly the problem is relaxed into a tight convex mixed-integer nonlinear problem (MINLP). Solving the convex relaxation of the developed model is much easier and would provide lower bound solution of the original problem. As compared to MILP-based global optimal solution algorithms for network design problem in the literature (e.g., Wang and Lo (2010); Liu and Wang (2015)), no additional binary variable are required to be introduced into the relaxed problem in this study, and thus the solution efficiency is significantly improved. Then, by embedding the solution of the relaxed problem as a lower bound, two iterative global optimization solution methods are proposed to obtain the global optimal solution of the developed model. One of the methods is a branch-and-bound based algorithm, which branches binary variables related to high flow links first to generate new sub-problems of the convex MINLP. Lower bound of the original problem is then updated from solving these sub-problems, so that the gap between lower bound and upper bound of the original problem could be reduced until certain requirement is fulfilled. The other method updates the lower bound problem by adding extra

constraints that force the removal of the obtained global optimal solution of the convex MINLP in previous iteration, so that the lower bound of the original problem would be updated at each iteration for unexplored possible set of road construction plan. Upper bound in both methods is derived by solving the SUE traffic assignment problem with the optimal construction plans of relaxed problems or sub-problems as input constant parameters. For both of the two proposed algorithms, global optimal solution is finally guaranteed when the difference between the lower bound and upper bound reach certain requirement. The proofs of the convergence of the two methods are then provided.

In summary, the discrete transportation network design problem with stochastic user equilibrium is addressed. This problem is a very common and important network design problem in transportation planning, however, has received little research attention in previous researches in the literature, primarily due to the computational difficulties (Chen and Alfa, 1991). In this chapter, global optimization solution algorithms are developed to solve the problem. Various linearization and convexification techniques are applied to formulate a convex relaxation of the original model, which provides the lower bound solution of the original problem and serves as the key component of two proposed global optimization solution algorithms.

This chapter is organized as follows: section 4.2 presents the model formulation, section 4.3 states the two global optimal solution methods including the model transformation and relaxation, section 4.4 reports the numerical study and section 4.5 concludes the chapter.

4.2 Model formulation

In this section, a discrete network design problem is proposed to determine the new link construction plan for the existing transportation network. The objective of the problem is to minimize the total system travel cost within a given budget while accounting for SUE traffic assignment principle, specifically, the logit route choice model. A mixed-integer nonlinear nonconvex model is formulated. It is assumed that the link travel time function follows the Bureau of Public Roads (BPR) equation, which is strictly increasing and convex.

The following notation is used for the model formulation:

Sets:

A	Link set of network, including candidate links
A_c	The set of candidate links to be constructed
A_r	The set of links that constitutes a path r
W	The set of origin-destination (O-D) pairs
R^w	The set of all existing paths and all potential paths that may exist after certain candidate links are constructed between O-D pair $w \in W$

Data and Parameters:

d^w	The travel demand between O-D pair $w \in W$
δ_{ar}^w	The link-path incidence, $\delta_{ar}^w=1$ if link $a \in A$ belongs to path $r \in R^w$ between O-D pair $w \in W$, otherwise $\delta_{ar}^w=0$
c_a	Construction cost of link $a \in A_c$
<i>Budget</i>	Given total construction budget for road network
T_a	Free flow travel time for link $a \in A$

y_a	Capacity of link $a \in A$, which usually defined as the maximum number of vehicles that can pass the road during a specific time period
R_a	The coefficient in BPR function that scale the rate at which congestion level affects the link travel time
M	A large enough positive number
θ	A positive parameter which scales the perceived travel time

Decision variables:

Z	The network total travel time
x_a	The traffic flow on link $a \in A$
u_a	Binary decision variable which equals to 1 if candidate link $a \in A_c$ is to be constructed and 0 otherwise
v_r	Decision variable which equals to 1 if a potential path $r \in R^w$ that includes any candidate link $a \in A_c$ exists and 0 otherwise; for those paths that consists solely of existing links, let $v_r = 1$
t_a	Travel time when using link $a \in A$, which is a function of link flow
h_r^w	Time of traversing path $r \in R^w$ between O-D pair $w \in W$
p_r^w	The probability of path $r \in R^w$ is selected, which can be given as explicit functions of link travel time
q_a^w	The probability of link $a \in A$ is chosen for a trip between O-D pair $w \in W$

The discrete network design model with stochastic user equilibrium can be formulated as the following mixed-integer nonlinear nonconvex programming:

[Original Problem]: nonconvex MINLP

$$\text{Minimize } Z = \sum_{a \in A} x_a \cdot t_a \quad (4.1)$$

Subject to

$$\sum_{a \in A_c} c_a u_a \leq \text{Budget} \quad (4.2)$$

$$u_a \in \{0,1\}, \quad a \in A_c \quad (4.3)$$

$$x_a = \sum_{w \in W} d^w \cdot q_a^w, \quad a \in A \quad (4.4)$$

$$q_a^w = \sum_{r \in R^w} \delta_{ar}^w \cdot p_r^w, \quad a \in A, \quad w \in W \quad (4.5)$$

$$p_{r_i}^w = \frac{v_{r_i} \cdot e^{-\theta h_{r_i}^w}}{\sum_{r_j \in R^w} v_{r_j} \cdot e^{-\theta h_{r_j}^w}}, \quad r_i \in R^w, \quad w \in W \quad (4.6)$$

$$h_r^w = \sum_{a \in A} \delta_{ar}^w \cdot t_a, \quad r \in R^w, \quad w \in W \quad (4.7)$$

$$v_r = \prod_{a \in A_r \cap A_c} u_a, \quad r \in R^w \text{ and } \{\text{links consist of } r\} \cap A_c \neq \emptyset \quad (4.8)$$

$$x_a \leq u_a \cdot M, \quad a \in A_c \quad (4.9)$$

$$t_a = T_a \cdot \left[1 + R_a \cdot \left(\frac{x_a}{y_a} \right)^4 \right], \quad a \in A \quad (4.10)$$

As was done in most previous NDP studies, the total system travel cost is adopted as the objective function, i.e., the problem is to seek the best construction plan which will optimize the system performance in terms of minimizing the total travel cost. Eq. (4.2) is the budget constraint which ensures that the total construction cost for adding new links will not exceed a certain amount of budget. Eq. (4.3) defines binary decision variables for candidate link addition, where $u_a=1$ infers the corresponding link being constructed and 0

otherwise. The set of candidate links are usually treated as given, predetermined by the local transport planners with consideration of limited budget, geographic restriction, economic development and even political issues. Eq. (4.4), (4.5) and (4.6) combine to enforce the stochastic user equilibrium traffic assignment principle, wherein Eq. (4.4) and (4.5) entail the relationship between link flow and path flow and the logit route choice model is described in Eq. (4.6). For this discrete network design problem, the existence of viable paths between each OD pair is determined by the link addition variable u_a . Indeed, the existence of certain viable path can be indicated by the variable v_r as defined in Eq. (4.8). If the multiplication of the binary variables u_a equals to 1, i.e., $v_r = 1$, it means all the links on this candidate path exist and this path is viable; otherwise, $v_r = 0$, which indicates that at least one of the links on this candidate path is not constructed and this path does not physically exist. Eq. (4.6) uses this variable v_r to enforce the logit split model, wherein v_r indicates that whether a candidate path exists or not. Eq. (4.7) defines the relationship between path travel time and link travel time, while the latter applies the form of typical BPR function as shown in Eq. (4.10). Eq. (4.9) imposes the restriction that traffic flows only exist on candidate links that will be constructed.

4.3 Global optimization solution method

As is well known, the nonconvex MINLP is hard to be solved even to its local optimum. Thus, traditional solution methods in the literature usually focused on obtaining a relatively good solution, while the global optimality of solutions cannot be guaranteed. However, in this chapter, two global optimal solution algorithms are proposed to solve the developed DNDP with SUE model by formulating a convexified relaxation of the original problem. The relaxed

problem is used to solve the lower-bound of the developed model, and the global optimal solution could be achieved through either a branch-and-bound based algorithm or an iterative method in which adding extra constraint in each iteration would update the lower bound solution. This section first elaborates the procedure of model transformation and relaxation, followed by the main idea and detailed steps of the two global optimal algorithms.

4.3.1 Model transformation and relaxation

4.3.1.1 Objective function

Substituting the link travel time function as defined in the form of BPR function (4.10), we have:

$$Z = \sum_{a \in A} T_a x_a + \sum_{a \in A} \frac{T_a R_a}{(y_a)^4} \cdot (x_a)^5 \quad (4.11)$$

It should be noted that the objective function is nonlinear but convex, as the road capacity for candidate link addition y_a is given and the only nonlinearity in (4.11) lies in the convex monomials of the second term.

4.3.1.2 Linear transformation of path existence decision variable

Eq.(4.8) is used to define the existence of potential paths, however, the multiplication of the binary variables makes the constraint hard to be tackled. Here, to remove the multiplication of binary variables, Eq. (4.8) is transformed into a set of linear constraints as below:

$$\begin{cases} v_r \geq 0 \\ v_r \leq u_a, \quad a \in A_r \cap A_c \\ v_r \geq \sum_{a \in A_r \cap A_c} u_a - (n_r - 1) \end{cases}, \quad r \in R^w \quad (4.12)$$

wherein n_r represents the total number of candidate links that this potential path r consists of. The equivalence between the set of constraints (4.12) and Eq. (4.8) under the condition of constraint (4.3) could be verified by separately letting $u_a = 1$ or 0. It is easy to deduce that the two constraints are indeed the same in case that each $u_a, a \in A_r \cap A_c$ is equal to 0 or each $u_a, a \in A_r \cap A_c$ is equal to 1. Otherwise, there will be at least one u_a equal to 0 and at least one u_a equal to 1. In this case, the second inequality of (4.12) will lead to $v_r \leq 0$; for the last inequality of (4.12), it is obvious that $\sum_{a \in A_r \cap A_c} u_a \leq n_r - 1$ and thus $\sum_{a \in A_r \cap A_c} u_a - (n_r - 1)$ is always non-positive. Combining the first constraint of $v_r \geq 0$ will result in $v_r \geq 0$. Therefore, we have $v_r = 0$ for this case, which also indicates the same result with Eq. (4.8). Hence, the equivalence between constraints (4.12) and Eq. (4.8) is proved under the condition of (4.3). By doing so, Eq. (4.8) is transformed into an equivalent set of constraints (4.12), which are completely linear constraints.

4.3.1.3 Linear relaxation of Logit route choice model

Since constraint (4.6) is a multi-dimensional nonlinear function, hence, it is not tractable of directly applying piecewise linearization technique on the constraint because the multi-dimensional arguments will result in a quite complicated mixed-integer programming for the approximation of (4.6). Aware of this problem, a series of transformation must be done before a piecewise linearization method can be easily applied, which means the nonlinear functions should be one-dimensional or two-dimensional. In the following transformation, the nonlinearity of (4.6) will be transferred into a group of one-dimensional logarithmic functions. In essence, the logit route choice model (4.6) employed

in the DNDP model entails two constraints: first, the route choice probability of existing path follows the logit split model and the probability is equal to zero for those non-existing paths; second, the summation of probabilities of all candidate paths for each OD pair equals to one. Let $p_r^w = v_r \cdot p_r^{\wedge w}$, (4.6) can be equivalently transformed into the following set of constraints:

$$\frac{p_{r_i}^{\wedge w}}{p_{r_j}^{\wedge w}} = \frac{e^{-\theta h_{r_i}^w}}{e^{-\theta h_{r_j}^w}}, \quad r_i, r_j \in R^w, \quad w \in W \quad (4.13)$$

$$\sum_{r \in R^w} p_r^w = 1, \quad w \in W \quad (4.14)$$

where $p_r^{\wedge w} \in [\underline{p_r^{\wedge w}}, \overline{p_r^{\wedge w}}]$ is a continuous and strictly positive variable; $\underline{p_r^{\wedge w}}$ is the lower bound of $p_r^{\wedge w}$ and can be set to a very small positive number, up to the specific accuracy requirement; $\overline{p_r^{\wedge w}}$ is the upper bound of $p_r^{\wedge w}$ and can be set to 1 because $p_r^{\wedge w}$ is a probability. $p_r^{\wedge w}$ equals to p_r^w if a path is existing, and has no restriction in its feasible domain otherwise. In other words, $p_r^{\wedge w} = p_r^w$ if $v_r = 1$ and $p_r^{\wedge w} \in [\underline{p_r^{\wedge w}}, \overline{p_r^{\wedge w}}]$ if $v_r = 0$. As this equation involves bilinear term $v_r \cdot p_r^{\wedge w}$, we want to apply the reformulation-linearization technique (Sherali and Adams, 1994, 1998) to convert this nonlinear relationship between $p_r^{\wedge w}$ and p_r^w into an equivalent set of linear constraints as shown below:

$$\begin{cases} p_r^w - v_r \cdot \underline{p_r^{\wedge w}} \geq 0 \\ p_r^w - v_r \cdot \overline{p_r^{\wedge w}} \leq 0 \\ p_r^w - p_r^{\wedge w} + \underline{p_r^{\wedge w}} - v_r \cdot \underline{p_r^{\wedge w}} \leq 0 \\ p_r^w - p_r^{\wedge w} + \overline{p_r^{\wedge w}} - v_r \cdot \overline{p_r^{\wedge w}} \geq 0 \end{cases} \quad (4.15)$$

By separately letting v_r be 1 or 0, we have,

$$v_r = 1 \Leftrightarrow \left\{ \begin{array}{l} \left\{ \begin{array}{l} p_r^w - \underline{p_r^w} \geq 0 \\ p_r^w - \overline{p_r^w} \leq 0 \end{array} \right\} \Leftrightarrow \underline{p_r^w} \leq p_r^w \leq \overline{p_r^w} \\ \left\{ \begin{array}{l} p_r^w - p_r^{\wedge w} \leq 0 \\ p_r^w - p_r^{\wedge w} \geq 0 \end{array} \right\} \Leftrightarrow p_r^w = p_r^{\wedge w} \end{array} \right\} \Leftrightarrow p_r^w = p_r^{\wedge w} \quad (4.16)$$

$$v_r = 0 \Leftrightarrow \left\{ \begin{array}{l} \left\{ \begin{array}{l} p_r^w \geq 0 \\ p_r^w \leq 0 \end{array} \right\} \Leftrightarrow p_r^w = 0 \\ \left\{ \begin{array}{l} p_r^w - p_r^{\wedge w} + \underline{p_r^w} \leq 0 \\ p_r^w - p_r^{\wedge w} + \overline{p_r^w} \geq 0 \end{array} \right\} \Leftrightarrow p_r^w = 0 \end{array} \right\} \quad (4.17)$$

Thus, $v_r \cdot p_r^{\wedge w} = p_r^w$ can be replaced by the equivalent linear constraints (4.15) and it can be proved via the deduction of (4.16) and (4.17).

As for the Eq. (4.13), it can be further reformulated into a simpler form by introducing a common scale variable β^w for each OD pair.

$$p_r^{\wedge w} = \beta^w \cdot e^{-\theta h_r^w}, \quad r \in R^w, \quad w \in W \quad (4.18)$$

By doing so, Eq. (4.18) is formed with a linear left-hand side and a nonlinear right-hand side. Furthermore, by adopting a continuous variable α^w to represent the scale variable β^w , $\beta^w = e^{\alpha^w}$, Eq. (4.18) can be transformed into the new constraint as below:

$$p_r^{\wedge w} = e^{-\theta h_r^w + \alpha^w}, \quad r \in R^w, \quad w \in W \quad (4.19)$$

So far, the simpler constraint (4.19) is equivalent to Eq. (4.18) and the original constraint (4.13). Note that both sides of (4.19) are strictly positive and the right-hand side is a two-variable nonlinear function. Taking logarithm on both sides of (4.19), we have:

$$\ln(\hat{p}_r^w) = -\theta h_r^w + \alpha^w, \quad r \in R^w, \quad w \in W \quad (4.20)$$

In the above constraint, the only nonlinearity is the logarithmic function on the left-hand side of (4.20), which is single-variable function and globally concave.

Figure 4.1 Polyhedral outer-approximation for $\ln(\hat{p}_r^w)$.

Now, the following part of this subsection provides a linear relaxation of the nonconvex constraint in (4.20) by constructing a polyhedral outer-approximation of logarithmic function $\ln(\hat{p}_r^w)$ over its feasible region $[\underline{p}_r^w, \overline{p}_r^w]$. Figure. 1 illustrates the mechanism of the employed polyhedral outer-approximation method, which consist of two tangential supports at the two given endpoints $\underline{p}_r^w$ and $\overline{p}_r^w$, and a curve chord that connects the two endpoints. Let $L_{p,r}^w$ denote $\ln(\hat{p}_r^w)$, the resulting linear relaxation for the logarithmic function is given as follows:

$$L_{p,r}^w \geq \ln(\underline{p_r^w}) + \frac{(p_r^w - \underline{p_r^w})}{(\overline{p_r^w} - \underline{p_r^w})} \left[\ln(\overline{p_r^w}) - \ln(\underline{p_r^w}) \right] \quad (4.21)$$

$$L_{p,r}^w \leq \ln(\underline{p_r^w}) + \frac{(p_r^w - \underline{p_r^w})}{\underline{p_r^w}} \quad (4.22)$$

$$L_{p,r}^w \leq \ln(\overline{p_r^w}) + \frac{(p_r^w - \overline{p_r^w})}{\overline{p_r^w}} \quad (4.23)$$

The right-hand side of (4.21) represents the curve chord between the endpoints and provide a lower bound constraint for the logarithmic function $\ln(p_r^w)$, whereas the tangential lines on the two endpoints supply two upper bound constraints and are described in constraints (4.22)-(4.23). Indeed, the original nonconvex logarithmic function is relaxed into a polyhedral region between the linear tangential lines and the curve chord.

In summary, constraints (4.15), (4.20) and (4.21)-(4.23) can be used to provide linear relaxation of logit model (4.6) in the original problem (Note that in (4.20), the logarithmic function $\ln(p_r^w)$ is replaced with the new variable $L_{p,r}^w$). As these relaxed constraints are linear, they can be directly included in the relaxed problem of the original model formulation, which could provide a lower bound of the original problem.

4.3.1.4 Convex relaxation of link travel time function

Although the link travel time function in the right-hand side of (4.10) is convex, this equation constraint is still nonconvex. However, it should be noted that this constraint is equivalent with the following two inequalities (4.24) and (4.25) with opposite inequality signs, one of which is convex and the other is concave.

$$t_a \geq T_a \cdot \left[1 + R_a \cdot \left(\frac{x_a}{y_a} \right)^4 \right], \quad a \in A \quad (4.24)$$

$$t_a \leq T_a \cdot \left[1 + R_a \cdot \left(\frac{x_a}{y_a} \right)^4 \right], \quad a \in A \quad (4.25)$$

Therefore, it is simple to relax this nonconvex constraint by retaining the convex inequality constraint (4.24) and removing the concave function (4.25). Indeed, as the objective of this problem is to minimize the total travel time, the removed constraint (4.25), which defines the upper constraints of travel time t_a , would not affect the optimal objective function value.

It should be noted that, to achieve the relaxation of constraint (4.10), one could also apply the same approach as demonstrated in Section 4.3.1.3, i.e., by constructing a linear outer-approximation based relaxation of this constraint.

In summary, the original problem is now relaxed into a convex MINLP as follows:

[Relaxed Problem]: convex MINLP

$$\text{Minimize } Z = \sum_{a \in A} T_a x_a + \sum_{a \in A} \frac{T_a R_a}{(y_a)^4} \cdot (x_a)^5 \quad (4.26)$$

Subject to

$$\sum_{a \in A_c} c_a u_a \leq \text{Budget} \quad (4.27)$$

$$u_a \in \{0,1\}, \quad a \in A_c \quad (4.28)$$

$$x_a = \sum_{w \in W} d^w \cdot q_a^w, \quad a \in A \quad (4.29)$$

$$q_a^w = \sum_{r \in R^w} \delta_{ar}^w \cdot p_r^w, \quad a \in A, \quad w \in W \quad (4.30)$$

$$L_{p,r}^w = -\theta h_r^w + \alpha^w, \quad r \in R^w, \quad w \in W \quad (4.31)$$

$$\begin{cases} L_{p,r}^w \geq \ln(\underline{p_r^w}) + \frac{(p_r^w - \underline{p_r^w})}{(\underline{p_r^w} - \underline{p_r^w})} \left[\ln(\overline{p_r^w}) - \ln(\underline{p_r^w}) \right] \\ L_{p,r}^w \leq \ln(\underline{p_r^w}) + \frac{(p_r^w - \underline{p_r^w})}{\underline{p_r^w}} \\ L_{p,r}^w \leq \ln(\overline{p_r^w}) + \frac{(p_r^w - \overline{p_r^w})}{\overline{p_r^w}} \end{cases}, \quad r \in R^w, \quad w \in W \quad (4.32)$$

$$\begin{cases} p_r^w - v_r \cdot \underline{p_r^w} \geq 0 \\ p_r^w - v_r \cdot \overline{p_r^w} \leq 0 \\ p_r^w - p_r^w + \underline{p_r^w} - v_r \cdot \underline{p_r^w} \leq 0 \\ p_r^w - p_r^w + \overline{p_r^w} - v_r \cdot \overline{p_r^w} \geq 0 \end{cases} \quad (4.33)$$

$$\sum_{r \in R^w} p_r^w = 1, \quad w \in W \quad (4.34)$$

$$h_r^w = \sum_{a \in A} \delta_{ar}^w \cdot t_a, \quad r \in R^w, \quad w \in W \quad (4.35)$$

$$\begin{cases} v_r \geq 0 \\ v_r \leq u_a, \quad a \in A_r \cap A_c \\ v_r \geq \sum_{a \in A_r \cap A_c} u_a - (n_r - 1) \end{cases}, \quad r \in R^w \quad (4.36)$$

$$x_a \leq u_a \cdot M, \quad a \in A_c \quad (4.37)$$

$$t_a \geq T_a \cdot \left[1 + R_a \cdot \left(\frac{x_a}{y_a} \right)^4 \right], \quad a \in A \quad (4.38)$$

Remark: The above convex MINLP problem is a relaxation of the original DNDP with SUE model. It should be noted that this relaxed problem has

convex objective function and constraints, except for the binary variables' constraint. (Indeed, only constraint (4.38) is nonlinear, but convex. All the other constraints are linear. As stated in section 4.3.1.4, if the same outer-approximation relaxation method as used in section 4.3.1.3 is applied, a set of completely linear constraints would be constructed here). Although it is not a completely convex problem due to the binary variables, the global optimization solution of convex MILP could be obtained by applying method like branch and bound. Many existing commercial software, like BONMIN, could be used to solve it. Solving this problem can provide a lower-bound solution of the original problem. This relaxation formulation avoids introducing additional binary variables and the MILP piecewise approximation into the relaxed problem, so that the high efficiency of the algorithm can be guaranteed.

4.3.2 Algorithm 1: a branch-and-bound based global optimal solution algorithm

In this subsection, a branch-and-bound based method is designed to update the convex MINLPs (i.e., Relaxed Problem) and find the global optimal solution of the OP within a specified tolerance of σ . The basic idea is to branch possible set of binary variable u_a to generate new sub-problems (convex MINLPs) with improved lower bound until the stopping criterion is fulfilled. As sub-problems can be directly solved to their global optimal solution by efficient optimization solver or conventional solution algorithm, their solutions can thus provide an updated and improved lower bound for the OP and a new candidate construction plan. By taking the obtained construction plan as input parameters of the transportation network and solving the traditional stochastic traffic assignment problem, we can obtain a feasible upper bound of the OP. More specifically, the node branching principle tends to branch the binary variables

u_a of those congested links first, which may have more significant impacts on the objective function, to form new convex MINLP sub-problems. This branching principle will produce less branch nodes and ensure finding the global optimal solution more quickly together with the developed tight RP.

4.3.2.1 Steps of the branch-and-bound based global optimal algorithm

Step 1: Initialization. Let $\sigma(\sigma > 0)$ be the maximum relative tolerance rate and denote the lower bound of the objective function value as $\underline{Z}$. The ideal value of σ is zero, while practically, a small value can be set upon specific requirement. Besides, set the initial upper bound $\bar{Z} = +\infty$. The active node set is defined as $S = \{s_0\}$, wherein s_0 is the root node of the binary tree and stands for the original feasible region of all variables, i.e., $s_0 = (u_a \in \{0, 1\}, a \in A_c; p_r^w \in [\underline{p_r^w}, \overline{p_r^w}], r \in R^w, w \in W)$.

Step 2: Lower bound and upper bound updating. Build and solve all sub-problems (relaxed problem) that are associated with all active nodes in S based on their feasible region separately. Find the minimum objective function value as the lower bound of the original problem, i.e., $\underline{Z} = \text{minimum}\{Z_s, s \in S\}$. Let s^* be the node related to the current lower bound $\underline{Z}$, and $u_s^*, s \in S$ be the obtained construction plan from each sub-problem. Using each u_s^* as input parameters and solving the traditional stochastic user equilibrium problem, the upper bound $\bar{Z} = \text{minimum}\{\bar{Z}, Z_s^{OP}, s \in S\}$ and the incumbent optimal solution u^* that relates to $\bar{Z}$ can be obtained, where Z_s^{OP} is the total system travel cost of construction plan u_s^* under SUE principle.

Step 3: Node fathoming. Whenever $Z_s \geq (1-\sigma)\bar{Z}, s \in S$ or the associated relaxed problem is infeasible, the node s is fathomed and should be eliminated from the active node set S , that is, all the nodes in set S should satisfy the condition $Z_s < (1-\sigma)\bar{Z}$.

Step 4: Stopping criterion. The procedure terminates if all nodes in the binary tree are fathomed and the active node set S is empty, or all the binary variables u_a are branched. List the incumbent optimal solution u^* , which is also the global optimal solution of the original problem; otherwise, go to step 5 and continue the solution procedure.

Step 5: Node branching. Select node s^* as the branching node and find the binary variable $u_{a^*}, a^* \in A_c$ with the largest link flow x_a among all those unbranched u_a and branch the node s^* by setting the selected u_a equal to one and zero separately, that is:

$$\begin{aligned} s_1 &= (u_{a,s1} = 0, \text{ when } a \in A_c \text{ and } a = a^*; \\ &\quad u_{a,s1} = u_{a,s^*}, \text{ when } a \in A_c \text{ and } a \neq a^*; \\ &\quad p_r^w \in [\underline{p_r^w}, \overline{p_r^w}], r \in R^w, w \in W) \end{aligned} \quad (4.39)$$

$$\begin{aligned} s_2 &= (u_{a,s2} = 1, a \in A_c \text{ and } a = a^*; \\ &\quad u_{a,s2} = u_{a,s^*}, a \in A_c \text{ and } a \neq a^*; \\ &\quad p_r^w \in [\underline{p_r^w}, \overline{p_r^w}], r \in R^w, w \in W) \end{aligned} \quad (4.40)$$

Substitute the new child nodes for s^* in the active set S and go to step 2 to continue the algorithm.

Proposition 1 (Convergence). The proposed algorithm 1 is convergent and, given any small enough tolerance $\sigma > 0$, the algorithm will terminate finitely with the incumbent solution being σ -global optimal to the OP.

Proof. Let Z_{s^*} be the objective function value of RP at node s^* and $\bar{Z}(u^*)$ be the best upper bound at the incumbent optimal solution u^* . With the foregoing branch-and-bound procedure, each new branch node will be checked if the active condition $Z_{s^*} > \bar{Z}(u^*)(1 - \sigma)$ is satisfied. Due to the limited number of candidate construction plans, the active node set S is always limited. The worst case of this algorithm is that all the new branch nodes are unfathomed and retained in S . The algorithm stops when all the binary variables are branched. The incumbent solution u^* , which associated to the best upper bound, can be regarded as the global optimal solution of the OP with the error between this solution and the real one is less than the given parameter σ . Since the relaxed problem is quite tight, it always provides a good lower bound to the original solution. Besides, the node branching principle tends to branch the binary variables of those probably congested links first, which have a significant impact on the objective function. The combination of both effects ensures less branches of the binary tree and higher speed in approaching to the stopping criterion.

4.3.3 Algorithm 2: an iterative global optimal solution algorithm based on adding extra constraints

Next, we will propose another global optimization solution algorithm, in which the updating of the lower and upper bounds of the objective function is achieved by introducing extra constraints in each iteration to remove the solved integer solution of last iteration from the full feasible set of construction plan

solutions (see, e.g., Wang et al. (2013)). We first solve the RP within the set of all possible road construction plan $\mathbf{u}$ and obtain the optimal solution $\mathbf{u}^*$. Then, by adding certain constraints to the RP to ensure that the obtained solutions of the RP (in terms of construction plan) from previous iteration can be excluded, we obtain a new optimal solution out of the unexplored set of solutions (in terms of construction plan), whose objective function value is updated (i.e., not smaller than the corresponding objective function values of all the obtained solutions in previous iterations). The incumbent optimal solution is also used as the input for the road network to calculate the total system travel time with a stochastic traffic assignment, which is indeed the upper bound of the original DNDP with SUE problem. If the upper bound corresponding to a generated solution is lower than the lower bound solution from previous iteration, then the generated solution is the global optimum, as the SUE based total travel time for any unexplored solution is not less than the obtained lower bound, which is no less than the minimum lower bound of all these explored solutions.

To clearly show how to expel the current solution and update the RP, let us see a small example first. Suppose we obtained an optimal construction plan $\mathbf{u}^* = [1 \ 0 \ 0 \ 1]$ in the last iteration, adding the following constraint (4.41) to the RP can directly exclude this solution from the feasible region and retain the other possible solutions, so that the lower bound of the OP could be updated and improved.

$$(1-u_1)+u_2+u_3+(1-u_4) \geq 1 \quad (4.41)$$

$\mathbf{u} = \mathbf{u}^*$ will lead to $(1-u_1)+u_2+u_3+(1-u_4)=0$ and thus violates constraint (4.41), whereas $(1-u_1)+u_2+u_3+(1-u_4) \geq 1$ always holds if $\mathbf{u} \neq \mathbf{u}^*$.

Extending this to general conditions, let U^* be the set of obtained solutions of the relaxed problem in each iteration, the following constraints are added to the relaxed problem in the next iteration:

$$\sum_{a \in A_c} \left[u_a^* (1 - u_a) + (1 - u_a^*) u_a \right] \geq 1, \quad u_a^* \in U^* \quad (4.42)$$

With the above additional constraint, all the obtained binary solutions will be expelled from the feasible region, and thus the new relaxed problem can be employed to update the lower and upper bounds of the objective function value.

4.3.3.1 Steps of the reducing-integer-solution based global optimal algorithm

Step 1: Initialization. Set the iteration number $iter = 0$, and the obtained relaxed problem solution $U^* = \emptyset$, the lower bound of the objective function value $\underline{Z} = 0$ and the upper bound $\bar{Z} = +\infty$.

Step 2: Solution of the relaxed problem. Set $iter = iter + 1$. Solve the relaxed problem with constraint (4.42). If the problem is feasible, let $u^{*,iter}$ be the obtained optimal solution of construction plan, $Z^{RP,iter}$ be the optimal objective function value of the RP and go to the next step. Otherwise (i.e., if the problem is infeasible), we have already obtained all the feasible solutions, and thus no other feasible solution exists, in which case the incumbent optimal solution u^* is the global optimal solution of the OP and the iterative process should be stopped.

Step 3: Update lower bound and upper bound of the objective function and the incumbent optimal solution. Update the lower bound of the objective function value of the DNDP model via $\underline{Z} = Z^{RP,iter}$. Solve the traditional SUE traffic

assignment problem by using $u^{*,iter}$ as input of the network construction plan and let the obtained total system travel time be $Z^{OP,iter}$. Update the upper bound of the objective function value of the DNDP model via $\bar{Z} = \min\{\bar{Z}, Z^{OP,iter}\}$. The incumbent optimal solution of the DNDP model is u^* , which is the optimal project associated with the current $\bar{Z}$.

Figure 4.2 An illustration example for Algorithm 2.

Step 4: Stopping criterion. Stop the iterative process if the incumbent $\underline{Z} \geq \bar{Z}$ and report the final global optimal solution u^* ; otherwise update $U^* = U^* \cup \{u^{*,iter}\}$ and go to Step 2 to continue the algorithm.

Before the detailed proof of the above-introduced global optimal solution algorithm is given, we schematically illustrate the rationale behind this algorithm by using Figure 4.2. In Figure 4.2, we have a limited number of

feasible solution of the network construction plan $\mathbf{u}$. At each specific plan u , we can solve the objective function value of the RP (denoted as Z^{RP} and represented by triangle shape in Figure 4.2) and the objective function value of the OP (denoted as Z^{OP} and represented by dot in Figure 4.2) while assuming this specific construction plan u is implemented. By solving RP in the first iteration, i.e. $iter = 1$, we obtain the global optimal solution $u^{*,1}$ for RP, which obviously has the lowest Z^{RP} for all the possible plan u , as is illustrated in Figure 4.2. Certainly, we also get the corresponding objective function value of RP $Z^{*,1}$ and OP $Z^{SUE,1}$, which are initialized as the lower bound and upper bound of the OP. Then, applying (4.42) to expel $u^{*,1}$ from the feasible region of RP, we have the second-best solution $u^{*,2}$ (as is shown in Figure 4.2, $u^{*,2}$ has the second lowest level of Z^{RP}) and also the associated objective function value. In each iteration, by removing all the obtained solutions (e.g., $\{u^{*,1}\}, \{u^{*,1}, u^{*,2}\}, \{u^{*,1}, u^{*,2}, u^{*,3}\}, \dots$) from the feasible domain of RP, we will sequentially obtain a new solution (e.g., $u^{*,2}, u^{*,3}, u^{*,4}, \dots$) and a pair of corresponding objective function values (Z^{RP} and Z^{OP}) for updating lower bound and upper bound. As the RP is a convex MINLP and extra constraint is added in each iteration, the obtained $Z^{*,iter}$, i.e., the lower bound, will gradually increase as the iterative process progresses. The iterative process will terminate when updated lower bound is larger than the incumbent upper bound. In this illustrative example of Figure 4.2, the iterative process stops at $u^{*,6}$, as the updated lower bound (i.e., the Z^{RP} at $u^{*,6}$, demonstrated by the yellow horizontal line) is greater than the incumbent upper bound (achieved at $u^{*,2}$ is Figure 4.2). In this case, the current upper bound will be the optimal solution of OP.

Proposition 2 (Convergence). The proposed global optimization algorithm 2 will terminate finitely with the incumbent solution being global optimal to the original problem.

Proof. On one hand, since the convex MINLP is a relaxed problem of the original model, the objective function value of the relaxed problem solution is no larger than that of the original model when the same binary solution of road construction plan is applied, i.e., $Z^{RP,1} \leq Z^{OP,1}$, $Z^{RP,2} \leq Z^{OP,2}$ ... $Z^{RP,iter} \leq Z^{OP,iter}$. On the other hand, the current global optimal solution of the RP will be excluded in the next iteration, which results in a new global optimal solution that is no better than the current one, that is, $Z^{RP,1} \leq Z^{RP,2} \leq \dots \leq Z^{RP,iter}$. Once the condition $\underline{Z} = Z^{RP,iter} \geq \bar{Z} = \min\{\bar{Z}, Z^{OP,iter}\}$ is satisfied, the iterative process should be stopped because the objective value of the original model in future iterations with SUE traffic assignment $Z^{SUE,iter+i}$ will be no better than $Z^{RP,iter}$, i.e., $Z^{RP,iter} \leq Z^{OP,iter}$, $Z^{RP,iter} \leq Z^{OP,iter+1}$, $Z^{RP,iter} \leq Z^{OP,iter+2}$ ... $Z^{RP,iter} \leq Z^{OP,iter+i}$. Hence, the global optimal solution of the original model exists among the set $\{Z^{OP,1}, Z^{OP,2}, \dots, Z^{OP,iter}\}$ and equal to the minimum value in this set, i.e. $Z^{OP} = \bar{Z}^{iter}$. Because the DNDP has a finite set of feasible solutions, the developed algorithm will terminate in a limited number of iterations. $\square$

Remark: The worst situation is to exclude all feasible solutions before the optimal solution is found. However, as the developed RP is very tight and removing the incumbent best integer solution greatly improves the lower bound of the OP, the solution efficiency of this algorithm is way better than the worst scenario of enumerating every feasible solution.

4.4 Numerical examples

To illustrate the application of the developed model and the performance of the proposed global optimal solution algorithm, we conduct two numerical examples in this section: one is a simple twelve-node network and the other is the widely used Sioux-Falls network. All numerical tests are performed 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 algorithms are coded in MATLAB R2012a, calling BONMIN (Bonami et al., 2008) through the free OPTI toolbox (Currie and Wilson, 2012) to solve convex MINLPs. The traditional SUE are solved by the Method of Successive Averages (MSA) (Sheffi, 1985), which is also coded with MATLAB.

4.4.1 Example 1: Simple twelve-node network

The twelve-node network, as shown in Figure 4.3, is the same with the network tested in Gao et al. (2005). It has twelve nodes and 23 links, among which link 1 to 6 are candidate links to be determined whether they should be constructed or not, and the other lines are existing links. There is one O-D pair in the network with node 1 and 12 as the origination and the destination respectively. The parameters used in the test network are all the same as used in Gao et al. (2005) and the link properties are marked in Figure 4.3. For the existing links, the number on the line represents the free flow travel time of each link; as for the candidate links, the first number in parentheses is the free flow travel time and the second number in parentheses stands for the construction cost of each link. The constant parameter θ in the logit model is assumed to be 0.1.

Figure 4.3 The twelve-node network.

Table 4.1 reports the results of the twelve-node network with different budgets, including the optimal projects, the corresponding objective function values and properties of algorithms. Numbers in the column of “optimal projects” stand for whether the six candidate links should be constructed: 1 means this candidate link should be constructed and 0 otherwise. The objective function value in the third column is indeed the total system travel time with traffic flows in SUE assignment on the network with the solved optimal project. Hence, it is reasonable that, with increasing given budget, the number of links to be constructed in optimal projects is going up, whereas the system travel time is decreasing. For algorithm 1, the branch-and-bound based method, number of nodes where the relaxed problem is calculated and CPU time in seconds are shown in Table 4.1, whereas for algorithm 2 the number of iterations is listed, as well as CPU time. Results of the test network show the efficiency of the two proposed global optimal algorithms, and it seems that algorithm 2 is overall a little bit better than algorithm 1. Since the test network size is not big, it is possible to use the enumeration method to verify the obtained global optimal solutions in Table 4.1. We enumerate all the potential network design plans

subject to a given budget and calculate the SUE distribution for each one. The optimal solutions are exactly the same as the results of the two algorithms, thus demonstrating that the proposed algorithm can solve to a global optimum.

Figure 4.4 shows the branching process of the proposed algorithm 1 when the given budget is 20. Node 1 is the initial node; a circle node indicates a feasible child node and a square child node is either infeasible or fathomed that should be removed from the active node set. Taking node 5 as an example, due to the objective value of the RP $Z_5 \geq (1 - 0.01)\bar{Z}$, the node is fathomed and should be eliminated from the active node set; taking node 7 as another example, the RP of node 7 is infeasible thus this node is also removed from the active node set. With progress of the solving process, the active set contains only one node (node 12) after all binary variables u are branched, and the stopping criteria is fulfilled. Hence, we can obtain $\mathbf{u}$ related to the current upper bound $\bar{Z}$ as the global optimal solution. When given budget is 40, since the lower bound and upper bound of the initial node satisfy the convergence condition immediately, the iterative process just stops at the initial. This indeed indicates the tightness of our developed RP to its OP.

In order to gain an insight of the proposed algorithm 2, iterative movement results of the solution process are demonstrated in Table 4.2, which includes the incumbent lower and upper bounds of the objective function value and the optimal results at each iteration. Without loss of generality, the cases of budget 20 and 40 are taken as testing examples. The values in the column of SUE are the corresponding network system travel time with SUE travel pattern under the current optimal construction plan. The minimum of this column is indeed the upper bound of the objective function value and the lower bounds are obtained by solving the convex MINLP in each iteration. The iterative process stops until

the current lower bound is larger than the upper bound. It should be noted that in some cases, e.g. when given the budget of 40, the final optimal result is even obtained in the first iteration.

4.4.2 Example 2: Sioux-Falls network

The well-known Sioux-Falls test network is adopted in this example. The Sioux-Falls network, which is shown in Figure 4.5, consists of 24 nodes and 38 pairs of links (each pair of links indicates a two-way road section). In this test example, the number of O-D pair, travel demands and link free flow travel time are set to be the same as those used in Suwansirikul et al. (1987). Other parameters and data applied in this example are listed in Appendix C. Five candidate links, marked with bold lines in Figure 4.5, are identified and we need to determine whether they should be constructed or not, with a total budget of 1570 ($10^3\$$). The lower bound of the path selection probability is set to 0.0001.

Table 4.3 shows the results of the DNDP with three different values of the parameter θ in the logit model. In all the three cases, the same optimal construction plan is obtained, i.e., adding links 16, 19 and 74 to the network. As the value of θ increasing, the objective function value is decreasing gradually and approaching to the DUE value, which is obtained via a traditional DUE traffic assignment principle assuming the optimal construction plan is implemented as input of the network. This is in line with the relationship between DUE and SUE mentioned above, i.e., SUE can include DUE as a special case. The CPU time in both algorithms is by and large decreasing as the θ goes up, which indicates that the presented model and algorithm can also be efficiently applied to approximate DNDP with DUE problem. Compared to Example 1 where algorithm 2 shows a slight advantage, in Example 2,

Table 4.1 Results of the test network under different budgets by the two proposed global optimal algorithms.

Budget	Optimal projects	Objective	Algorithm 1		Algorithm 2	
			No. of nodes	CPU Time (s)	No. of iterations	CPU Time (s)
20	1 0 1 0 0 0	3971.6799	7	20.3	3	5.7
40	1 1 0 0 0 1	2540.8144	1	4.5	3	7.4
60	1 1 1 0 1 1	2381.8727	7	50.6	4	15.1

Figure 4.4 The binary tree of the proposed Algorithm 1 when budget=20.

Table 4.2 Iterative process of the proposed Algorithm 2.

Budget	Iteration	Optimal results	SUE	Lower bound	Upper bound	Final result
20	1	1 1 0 0 0	4020.2156	3910.2067	4020.2156	N
	2	1 0 1 0 0	3971.6799	3916.8270	3971.6799	Y
	3	1 0 0 1 0	4085.4009	3991.5561	3971.6799	N
40	1	1 1 0 0 0 1	2540.8144	2475.0100	2540.8144	Y
	2	1 0 0 0 1 1	2561.0137	2482.5266	2540.8144	N
	3	1 0 1 0 0 1	2628.3655	2554.4760	2540.8144	N

Figure 4.5 Sioux-Falls network.

Table 4.3 Results of Sioux-Falls network with three different θ value adopted in logit model.

θ	Optimal projects	Objective	CPU time of Algorithm 1 (min)	CPU time of Algorithm 2 (min)
DUE	16,19,74	132.7324	-	-
1000	16,19,74	132.6171	6.1	10.3
30	16,19,74	136.5458	9.5	10.0
15	16,19,74	146.8552	10.0	17.4

algorithm 1 performs a little better than algorithm 2. It might be because the performance of the two algorithms related to the topology of the specific network. In all, this numerical test, on one hand, demonstrates the solution efficiency of the developed solution methods to a medium-size network problem; on the other hand, it justifies the consideration of the more general SUE in network design problem, as it could be used to approximate the case of DUE (i.e., it can include the network design problem with DUE as a special case) if the parameter θ is set sufficiently large.

4.5 Concluding remarks

In this chapter, a discrete network design problem with stochastic user equilibrium traffic assignment is studied. A mixed-integer nonlinear nonconvex model is solved to its global optimum by two presented global optimization solution methods. Computational examples are tested to illustrate the implementation of the solution algorithms and the results show efficiency of the proposed methods. Instead of using MILP to approximate the nonlinear functions in the original model, which is the spirit of some previous global optimal solution algorithms for continuous network design problem, a convex mixed-integer nonlinear programming is applied to relax the original model. Also different from most of the MILP based methods, an exact global optimal

solution of the original model rather than only an approximated global optimal solution could be guaranteed via the proposed global optimal solution algorithms.

Chapter 5 A NOVEL DISCRETE NETWORK DESIGN PROBLEM FORMULATION AND ITS GLOBAL OPTIMIZATION SOLUTION ALGORITHM

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. This chapter addresses 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.

5.1 Introduction

In previous studies, the discrete network design problems assume predetermined 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 chapter, a DNDP problem is exploited 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 chapter). 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 chapter, and this study could contribute in filling 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, a Reformulation-linearization technique (Sherali and Adams, 1994, 1998) is applied 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. 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 nonconvex, 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 programming 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 essence, this chapter contributes to the state-of-the-art 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 the author's 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 Luatkep 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 chapter is organized as follows. Section 5.2 presents the original model formulation and the relaxed mixed-integer linear model reformulation. Section 5.3 proposes the global optimal algorithm. Section 5.4 discusses several practical considerations. Section 5.5 reports numerical examples. The final section summarizes the chapter.

5.2 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, $A = A_1 \cup A_2$
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.
Δ	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, $\mathbf{u} = [u_a]$, $a \in A$. 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 γ_a in subsection 5.2.2.1) 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]

$$\min_{x,y,u} Z_{\text{OP}} = \sum_{a \in A_1} x_a t_a(x_a) + \sum_{a \in A_2} x_a t_a(x_a, y_a, u_a) \quad (5.1)$$

Subject to:

$$\underline{y}_a \leq y_a \leq \bar{y}_a, \quad \forall a \in A_2 \quad (5.2)$$

$$\sum_{a \in A_2} g_a(y_a, u_a) = \sum_{a \in A_2} (\alpha u_a y_a + \beta u_a) \leq B \quad (5.3)$$

$$x_a \leq u_a M, \quad \forall a \in A_2 \quad (5.4)$$

$$u_a \in \{0, 1\}, \quad \forall a \in A_2 \quad (5.5)$$

$$\mathbf{x} = \mathbf{x}^*(\mathbf{y}, \mathbf{u}) \quad (5.6)$$

$$t_a(x_a) = T_a \left(1 + R_a \left(\frac{x_a}{Y_a} \right)^4 \right), \quad \forall a \in A_1 \quad (5.7)$$

$$t_a(x_a, y_a, u_a) = T_a \left(1 + R_a \left(\frac{x_a}{y_a} \right)^4 \right) + (1 - u_a)M, \quad \forall a \in A_2 \quad (5.8)$$

The objective function of this formulation in Eq. (5.1) is the total travel time cost from both existing links and candidate links. Constraint (5.2) expresses the

restriction of candidate road capacity. Budgetary constraint (5.3) entails that the total construction cost is less than the maximum allowable expenditure for network improvement. In constraint (5.3), the second term βu_a indicates the fixed cost of new road, that is, the fixed cost β 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 (5.4) and (5.5) 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 (5.6) enforces the flow pattern with Deterministic User Equilibrium, where $\mathbf{x}^*(\mathbf{y}, \mathbf{u})$ is the vector of DUE flows for given vector of link capacities $\mathbf{y}$ and vector of binary decision variables $\mathbf{u}$. Constraints (5.7) and (5.8) use the typical BPR function to define the link travel time. In (5.7), the travel times for existing links only depend on the travel flows x_a as the link capacities Y_a are given. In constraint (5.8), 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 (5.8) 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 (5.3) and (5.4) ensure zero traffic flow on unconstructed link and therefore the term $x_a t_a(x_a, y_a, u_a)$ 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 (5.3) and BPR travel time function in constraints (5.7) and (5.8). In designing global optimization solution method for this model, different techniques are applied to deal with the two types of nonlinear terms.

5.2.1 Variational inequality function of traffic assignment problem

As is mentioned in the last section, in this chapter, 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 $(\mathbf{y}, \mathbf{u})$, the VI problem is to find the optimal solution $\mathbf{x}^* \in \Psi$ which satisfies the following constraints

$$\sum_{a \in A_1} t_a(x_a^*) \cdot (x_a^* - x_a) + \sum_{a \in A_2} t_a(x_a^*, y_a, u_a) \cdot (x_a^* - x_a) \leq 0, \quad \forall x_a \in \Psi, \quad (5.9)$$

$$\Psi = \left\{ \mathbf{x} \mid x_a = \sum_{w \in W} v_a^w, \Delta \cdot \mathbf{v}^w = \mathbf{d}^w, v_a^w \geq 0, \forall a \in A, w \in W \right\}, \quad (5.10)$$

where Ψ is a feasible set of traffic flow on the network.

Since all the constraints in Ψ are linear, Ψ is actually a bounded polytope.

Let C be the indices 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,

$$\mathbf{x} = \sum_{c \in C} \lambda_c \mathbf{x}^c, \quad (5.11)$$

$$\sum_{c \in C} \lambda_c = 1, \quad 0 \leq \lambda_c \leq 1, \quad \forall c \in C, \quad (5.12)$$

where λ_c is the weighted factor of the c th corner-point $\mathbf{x}^c$ of the polytope Ψ .

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 (5.9)-(5.10) if and only if $\mathbf{x}^*$ satisfies the following problem

$$\sum_{a \in A_1} t_a(x_a^*) \cdot (x_a^* - x_a^c) + \sum_{a \in A_2} t_a(x_a^*, y_a, u_a) \cdot (x_a^* - x_a^c) \leq 0, \quad \forall c \in C \quad (5.13)$$

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

In conclusion, Eq. (5.14) can be formulated to stand for the VI problem of the DUE condition.

$$\sum_{a \in A_1} t_a(x_a^*) \cdot (x_a^* - x_a^c) + \sum_{a \in A_2} [t_a(x_a^*, y_a) + (1 - u_a)M] \cdot (x_a^* - x_a^c) \leq 0, \quad \forall c \in C \quad (5.14)$$

where $x_a^* \in \Psi$. $\square$

5.2.2 Reformulation of multivariate polynomial function

In this section, two types of nonlinear terms are dealt with, i.e., the multivariate link travel time functions and the bilinear functions. The link travel time functions will be reformulated into logarithmic 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 (5.8). 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.

5.2.2.1 Link travel time function

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

$$t_a(x_a, y_a) = T_a \left(1 + R_a \left(\frac{x_a}{y_a} \right)^4 \right), \quad \forall a \in A \quad (5.15)$$

where T_a is the 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.

Letting a new variable h_a represent the monomial $(x_a)^4 / (y_a)^4$ as in (5.15), we have:

$$h_a = \left(\frac{x_a}{y_a} \right)^4, \quad \forall a \in A \quad (5.16)$$

Since $x_a \geq 0, \forall a \in A$, the logarithm of both sides of Eq. (5.16) cannot be simply taken. 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 γ_a ($\gamma_a \in \{0,1\}, \forall a \in A$) are introduced for each link $a \in A$. Let

$$x_a = \gamma_a \tilde{x}_a, \quad \forall a \in A \quad (5.17)$$

$$\tilde{h}_a = \left(\frac{\tilde{x}_a}{y_a} \right)^4, \quad \forall a \in A \quad (5.18)$$

Thus, by substituting Eq. (5.17) into Eq. (5.16), the following Eq. (5.19) can be induced:

$$h_a = \gamma_a \tilde{h}_a, \quad \forall a \in A \quad (5.19)$$

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 (5.4).

If $u_a^* = 1$ 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 $(x_a^*, y_a^*, u_a^*, \gamma_a^*, h_a^*)$, $u_a^* = 1, \gamma_a^* = 0$ 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 $u_a^* = 1, \gamma_a^* = 0$ is true in one's optimal investment plan, which means a new link addition is completely not used in the network, this new link can be decided not to be constructed. 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, the construction cost is reduced, making the budget constraint still fulfilled. That is to say, even in optimal solutions, $u_a^* = 1, \gamma_a^* = 0$, an equivalent optimal solution (i.e., with the same objective value) can be obtained 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 $a \in A_1$, γ_a only indicates whether traffic flow on this link is zero or not. To simplify the denotation, the binary variable u_a ($\forall a \in A_1$) is also used to represent γ_a of existing links, which results in:

$$x_a = u_a \tilde{x}_a, \quad \forall a \in A \quad (5.20)$$

$$h_a = u_a \tilde{h}_a, \quad \forall a \in A \quad (5.21)$$

Taking logarithm on both sides of Eq. (5.18), we have:

$$\ln \tilde{h}_a = 4 \ln \tilde{x}_a - 4 \ln y_a, \quad \forall a \in A \quad (5.22)$$

So far, the monomial in the BPR function is transformed into Eq. (5.22), wherein the nonlinearity is only contained in the logarithmic functions. That is,

other than these logarithmic functions, Eq. (5.22) is in fact in linear form.

Letting $L_{ha} = \ln(\tilde{h}_a)$, $L_{xa} = \ln(\tilde{x}_a)$ and $L_{ya} = \ln(y_a)$, we have

$$L_{ha} = 4L_{xa} - 4L_{ya}, \quad \forall a \in A \quad (5.23)$$

The link travel cost function can be replaced by:

$$t_a(x_a, y_a) = T_a + T_a R_a h_a, \quad \forall a \in A \quad (5.24)$$

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.

5.2.2.2 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.

$$\sum_{a \in A} x_a \cdot t_a(x_a, y_a) = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a \frac{(x_a)^5}{(y_a)^4} \quad (5.25)$$

Following the same technique introduced in the last subsection, the new variable p_a is used to replace the monomial part in Eq. (5.25):

$$p_a = \frac{(x_a)^5}{(y_a)^4}, \quad \forall a \in A \quad (5.26)$$

As is done above, substituting Eq. (5.20) into Eq. (5.26) and introducing a new continuous variable $\tilde{p}_a$ ($0 < \tilde{p}_a \leq M, \forall a \in A$) leads to:

$$\tilde{p}_a = \frac{(\tilde{x}_a)^5}{(y_a)^4}, \quad \forall a \in A \quad (5.27)$$

$$p_a = u_a \tilde{p}_a, \quad \forall a \in A \quad (5.28)$$

Taking logarithm on both sides of Eq. (5.27) leads to:

$$\ln(\tilde{p}_a) = 5\ln(\tilde{x}_a) - 4\ln(y_a), \quad \forall a \in A \quad (5.29)$$

Similarly, by letting L_{pa} to stand for $\ln(\tilde{p}_a)$, Eq. (5.29) is rewritten as

$$L_{pa} = 5L_{xa} - 4L_{ya}, \quad \forall a \in A \quad (5.30)$$

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

$$\sum_{a \in A} x_a \cdot t_a(x_a, y_a) = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a \quad (5.31)$$

In this case, the objective function can be represented by Eq. (5.31) because it is exactly the total system travel time.

As for the VI constraints, by plugging Eq. (5.31) into Eq. (5.14), we have:

$$\sum_{a \in A} T_a x_a^* + \sum_{a \in A} T_a R_a p_a + \sum_{a \in A_2} (x_a^* - u_a x_a^*) M - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C \quad (5.32)$$

In Eq. (5.32), 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. (5.32) can be simplified into a linear constraint:

$$\sum_{a \in A} T_a x_a^* + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C \quad (5.33)$$

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

5.2.3 Linear transformation of bilinear function using RLT technique

In the model formulation, bilinear terms are also involved. In this subsection, a linearization technique will be applied 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. (5.20) $x_a = u_a \tilde{x}_a$, $\forall a \in A$ as an example. It is supposed that $\underline{x}_a \leq \tilde{x}_a \leq \bar{x}_a$, where $\underline{x}_a$ and $\bar{x}_a$ 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]

$$\begin{cases} x_a - u_a \underline{x}_a \geq 0 \\ x_a - u_a \bar{x}_a \leq 0 \\ x_a - \tilde{x}_a + \underline{x}_a - u_a \underline{x}_a \leq 0 \\ x_a - \tilde{x}_a + \bar{x}_a - u_a \bar{x}_a \geq 0 \end{cases}, \quad \forall a \in A \quad (5.34)$$

Eq. (5.20) can be directly rewritten as "if-and-only-if" conditions, which are represented as:

$$\begin{cases} x_a = 0 \Leftrightarrow u_a = 0 \\ x_a = \tilde{x}_a \Leftrightarrow u_a = 1 \end{cases}, \quad \forall a \in A \quad (5.35)$$

Therefore, by separately substituting two possible values of u_a into Eq. (5.34), we have:

$$u_a = 0 \Leftrightarrow \begin{cases} x_a \geq 0 \\ x_a \leq 0 \\ x_a - \tilde{x}_a + \underline{x}_a \leq 0 \\ x_a - \tilde{x}_a + \bar{x}_a \geq 0 \end{cases} \Leftrightarrow \begin{cases} x_a = 0 \\ \underline{x}_a \leq \tilde{x}_a \leq \bar{x}_a \end{cases} \quad (5.36)$$

$$u_a = 1 \Leftrightarrow \begin{cases} x_a - \underline{x}_a \geq 0 \\ x_a - \bar{x}_a \leq 0 \\ x_a - \tilde{x}_a \leq 0 \\ x_a - \tilde{x}_a \geq 0 \end{cases} \Leftrightarrow \begin{cases} x_a = \tilde{x}_a \\ \underline{x}_a \leq x_a \leq \bar{x}_a \end{cases} \quad (5.37)$$

The above result shows that Eq. (5.34) is identical to the "if-and-only-if" condition in Eq. (5.35). Thus, equivalence between Eq. (5.34) and Eq. (5.20) is also verified. This linear transformation can be used 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 $\underline{h}_a \leq \tilde{h}_a \leq \bar{h}_a$ and $\underline{p}_a \leq \tilde{p}_a \leq \bar{p}_a$, it is convenient to implement the RLT technique to obtain the equivalent linearization of Eq. (5.21) and Eq. (5.28).

It should be noted that the construction cost function $g_a(u_a, y_a)$ also involves a bilinear term in constraint (5.3).

Considering that the upper bound and lower bound of y_a are already given in the road capacity restriction in Eq. (5.2), 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:

$$g_a(u_a, y_a) = \alpha k_a + \beta u_a, \quad \forall a \in A_2 \quad (5.38)$$

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 $\bar{d}$ 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]

$$\min_{\mathbf{x}, \mathbf{y}, \mathbf{u}} Z_{MINLP} = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a \quad (5.39)$$

Subject to:

$$\underline{y}_a \leq y_a \leq \bar{y}_a, \quad \forall a \in A_2 \quad (5.40)$$

$$\sum_{a \in A_2} (\alpha u_a y_a + \beta u_a) \leq B \quad (5.41)$$

$$x_a \leq u_a M, \quad \forall a \in A \quad (5.42)$$

$$\sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C \quad (5.43)$$

$$t_a = T_a + T_a R_a h_a, \quad \forall a \in A \quad (5.44)$$

$$L_{ha} = 4L_{xa} - 4L_{ya}, \quad \forall a \in A \quad (5.45)$$

$$L_{ha} = \ln(\tilde{h}_a), \quad \forall a \in A \quad (5.46)$$

$$L_{xa} = \ln(\tilde{x}_a), \quad \forall a \in A \quad (5.47)$$

$$L_{ya} = \ln(y_a), \quad \forall a \in A \quad (5.48)$$

$$L_{pa} = 5L_{xa} - 4L_{ya}, \quad \forall a \in A \quad (5.49)$$

$$L_{pa} = \ln(\tilde{p}_a), \quad \forall a \in A \quad (5.50)$$

$$\left. \begin{aligned} \hat{d} - u_a \underline{d} &\geq 0 \\ \hat{d} - u_a \bar{d} &\leq 0 \\ \hat{d} - d + \underline{d} - u_a \underline{d} &\leq 0 \\ \hat{d} - d + \bar{d} - u_a \bar{d} &\geq 0 \\ d &\in D = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\} \end{aligned} \right\} \quad (5.51)$$

$$u_a \in \{0, 1\}, \quad \forall a \in A \quad (5.52)$$

$$\Psi = \left\{ \mathbf{x} \mid x_a = \sum_{w \in W} v_a^w, \Delta \cdot \mathbf{v}^w = \mathbf{d}^w, v_a^w \geq 0, \forall a \in A, w \in W \right\} \quad (5.53)$$

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

5.3 Solution algorithm

5.3.1 Model relaxation

5.3.1.1 Relaxation of general logarithmic function

One can find that the nonlinearity of the above shown MINLP only lies in the four logarithmic functions: $\ln(\tilde{x}_a)$, $\ln(y_a)$, $\ln(\tilde{h}_a)$ and $\ln(\tilde{p}_a)$. 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 $L_{xa} = \ln(\tilde{x}_a)$ 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 $[\underline{x}_a, \bar{x}_a]$. The interval is divided into $N-1$ small segments by selecting $N-2$ breakpoints between the two endpoints $\underline{x}_a$ and $\bar{x}_a$. The series of breakpoints and two endpoints are denoted by x_a^n , $\forall n \in 1, 2, \dots, N$. It should be noted that there is no need to partition the interval into equal segments. Similar as shown in Figure 3.1, the linear relaxation of the concave logarithmic function $\ln(\tilde{x}_a)$ 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.

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

[LR: Linear Relaxation]

$$L_{xa} \leq \ln(x_a^n) - 1 + \frac{\tilde{x}_a}{x_a^n}, \quad \forall x_a^n = \underline{x}_a + \frac{\bar{x}_a - \underline{x}_a}{N-1} \cdot (n-1), \quad n=1, 2, \dots, N \quad (5.54)$$

$$\sum_{n=1}^N \theta_{xa}^n x_a^n = \tilde{x}_a \quad (5.55)$$

$$\sum_{n=1}^N \theta_{xa}^n \ln(x_a^n) \leq L_{xa} \quad (5.56)$$

$$\sum_{n=1}^N \theta_{xa}^n = 1 \quad (5.57)$$

$$\theta_{xa}^n \geq 0, \quad n=1, 2, \dots, N \quad (5.58)$$

$$\theta_{xa}^n \leq \lambda_{xa}^{n-1} + \lambda_{xa}^n, \quad n=2, 3, \dots, N-1, \quad \theta_{xa}^1 \leq \lambda_{xa}^1, \quad \theta_{xa}^N \leq \lambda_{xa}^{N-1} \quad (5.59)$$

$$\sum_{n=1}^{N-1} \lambda_{xa}^n = 1 \quad (5.60)$$

$$\lambda_{xa}^n \in \{0, 1\}, \quad n=1, 2, \dots, N-1 \quad (5.61)$$

The upper bound of $\ln(\tilde{x}_a)$ is given in Eq. (5.54), the right-hand side of which denotes all the tangent lines on each point, whereas the lower bound is provided by Eqs. (5.55)-(5.61), which represents all the piecewise linear interpolations between each pair of consecutive points.

To formulate the piecewise linear function, N continuous variables θ_{xa}^n , $n=1, 2, \dots, N$ and $N-1$ binary variables λ_{xa}^n , $n=1, 2, \dots, N-1$ are introduced. The binary variable λ_{xa}^n indicates whether an interval is active or not, that is: $\tilde{x}_a$ falls in this interval $[x_a^n, x_a^{n+1}]$ if $\lambda_{xa}^n = 1$ and $\tilde{x}_a \notin [x_a^n, x_a^{n+1}]$ if $\lambda_{xa}^n = 0$. The continuous variables θ_{xa}^n , $n=1, 2, \dots, N$ are the coefficients associated with each breakpoint and measure the location of $\tilde{x}_a$ between the two endpoints of the active interval: specifically, $\theta_{xa}^n = (\tilde{x}_a - x_a^n) / (x_a^{n+1} - x_a^n)$

and $\theta_{xa}^{n+1} = (x_a^{n+1} - \tilde{x}_a) / (x_a^{n+1} - x_a^n)$ 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 $\tilde{x}_a$ falls within the active interval $[x_a^n, x_a^{n+1}]$, Eq.

(5.60) guarantees that only λ_{xa}^n is equal to 1 and all the other

λ_{xa}^m , $m = 1, \dots, n-1, n+1, \dots, N-1$ are equal to 0. According to Eqs. (5.57)-(5.59),

it implies that $\theta_{xa}^n + \theta_{xa}^{n+1} = 1$, $\theta_{xa}^n \geq 0$, $\theta_{xa}^{n+1} \geq 0$ and $\theta_{xa}^m = 0$,

$\forall m = 1, \dots, n-1, n+2, \dots, N$. Hence, $\tilde{x}_a$ can be represented by a convex

combination from Eq. (5.55), i.e. $\tilde{x}_a = \theta_{xa}^n x_a^n + \theta_{xa}^{n+1} x_a^{n+1}$, and the lower bound of

$\ln(\tilde{x}_a)$ can be evaluated from Eq. (5.56), i.e. $\theta_{xa}^n \ln(x_a^n) + \theta_{xa}^{n+1} \ln(x_a^{n+1}) \leq L_{xa}$.

Combined with Eq. (5.54), the feasible region

$\theta_{xa}^n \ln(x_a^n) + \theta_{xa}^{n+1} \ln(x_a^{n+1}) \leq L_{xa} \leq \ln(x_a^n) - 1 + \tilde{x}_a / x_a^n$ 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.

5.3.1.2 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 Ψ . 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:

$$\sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C_s, \quad C_s \subseteq C \quad (5.62)$$

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]

$$\min_{x, y, u} Z_{R-MINLP} = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a \quad (5.63)$$

Subject to the same constraints (5.40)-(5.42), (5.44)-(5.53), and the relaxed VI constraints (5.62).

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 D).

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, let Q stand for the set of variables, whose logarithmic functions need linear relaxation. Thus, for arbitrary variable $q \in Q$, $\underline{q}$ and $\bar{q}$ are the lower and upper bounds

respectively; L_q denotes the logarithmic function $\ln(q)$, for example, L_q actually represents L_{y_a} if $q = y_a$. In conclusion, the R-MILP can be expressed as follows:

[R-MILP: Relaxed MILP]

$$\min_{x,y,u,t} Z_{R-MILP} = \sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a \quad (5.64)$$

Subject to:

$$\underline{y}_a \leq y_a \leq \bar{y}_a, \quad \forall a \in A_2 \quad (5.65)$$

$$\sum_{a \in A_2} (\alpha u_a y_a + \beta u_a) \leq B \quad (5.66)$$

$$x_a \leq u_a M, \quad \forall a \in A \quad (5.67)$$

$$\sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a^c - \sum_{a \in A_2} (1 - u_a) M \cdot x_a^c \leq 0, \quad \forall c \in C_s, \quad C_s \subseteq C \quad (5.68)$$

$$t_a = T_a + T_a R_a h_a, \quad \forall a \in A \quad (5.69)$$

$$L_{ha} = 4L_{xa} - 4L_{ya}, \quad \forall a \in A \quad (5.70)$$

$$L_{pa} = 5L_{xa} - 4L_{ya}, \quad \forall a \in A \quad (5.71)$$

$$\left. \begin{aligned} \hat{d} - u_a \underline{d} &\geq 0 \\ \hat{d} - u_a \bar{d} &\leq 0 \\ \hat{d} - d + \underline{d} - u_a \underline{d} &\leq 0 \\ \hat{d} - d + \bar{d} - u_a \bar{d} &\geq 0 \\ d &\in D = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\} \end{aligned} \right\} \quad (5.72)$$

$$u_a \in \{0, 1\}, \quad \forall a \in A \quad (5.73)$$

$$\Psi = \left\{ \mathbf{x} \mid x_a = \sum_{w \in W} v_a^w, \Delta \cdot \mathbf{v}^w = \mathbf{d}^w, v_a^w \geq 0, \forall a \in A, w \in W \right\} \quad (5.74)$$

$$\left. \begin{aligned}
& L_q \leq \ln(q^n) - 1 + \frac{q}{q^n}, \quad \forall q^n = \underline{q} + \frac{\bar{q} - \underline{q}}{N-1} \cdot (n-1), \quad n = 1, 2, \dots, N \\
& \sum_{n=1}^N \theta_q^n q^n = q \\
& \sum_{n=1}^N \theta_q^n \ln(q^n) \leq L_q \\
& \sum_{n=1}^N \theta_q^n = 1 \\
& \theta_q^n \geq 0, \quad n = 1, 2, \dots, N \\
& \theta_q^n \leq \lambda_q^{n-1} + \lambda_q^n, \quad n = 2, 3, \dots, N-1, \quad \theta_q^1 \leq \lambda_q^1, \quad \theta_q^N \leq \lambda_q^{N-1} \\
& \sum_{n=1}^{N-1} \lambda_q^n = 1 \\
& \lambda_q^n \in \{0, 1\}, \quad n = 1, 2, \dots, N-1
\end{aligned} \right\} \quad (5.75)$$

$$q \in Q = \{\tilde{x}_a, \tilde{h}_a, \tilde{p}_a, \forall a \in A; y_a, \forall a \in A_2\} \quad (5.76)$$

5.3.2 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.

5.3.2.1 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 is not excluded. In this way, with only 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 $x \in X_{\text{var}}$, where X_{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]

$$\underline{x}^{\text{new}} = \text{Min } x \quad (5.77)$$

subject to:

$$\underline{x}^{\text{old}} \leq x \leq \bar{x}^{\text{old}}, \quad \forall x \in X_{\text{var}} \quad (5.78)$$

$$Z_{R\text{-MILP}} \leq \bar{Z}_{\text{MINLP}} \quad (5.79)$$

All the other constraints in the R-MILP model except bound constraints. (5.80)

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

$$\bar{x}^{\text{new}} = \text{Max } x \quad (5.81)$$

subject to:

$$\underline{x}^{\text{old}} \leq x \leq \bar{x}^{\text{old}}, \quad \forall x \in X_{\text{var}} \quad (5.82)$$

$$Z_{R\text{-MILP}} \leq \bar{Z}_{\text{MINLP}} \quad (5.83)$$

All the other constraints in the R-MILP model except bound constraints. (5.84)

where $\underline{x}^{\text{old}}$ and $\bar{x}^{\text{old}}$ are respectively the current lower and upper bounds for variable $x \in X_{\text{var}}$ before update, $\bar{Z}_{\text{MINLP}}$ is a known upper bound of the global optimal objective function value of the original MINLP model. $\bar{Z}_{\text{MINLP}}$ 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 $\bar{Z}_{MINLP}$. It is worth noted that the set of variables that needs bounds update should be carefully selected, because calculating new bounds also influences 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.

5.3.2.2 Global optimization solution algorithm

Based on the above analysis, a global optimization solution algorithm is developed 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, the framework of this solution approach is shown in Figure 5.1. 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 with guarantee the global optimal solution remains 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.

Figure 5.1 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 ζ^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 MINLP 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 D) with σ^i and solve it to obtain $\mathbf{x}'^*$. Check whether the condition

$$\sum_{a \in A} T_a x_a^\sigma + \sum_{a \in A} T_a R_a p_a^\sigma - \sum_{a \in A} t_a^\sigma x_a'^* - \sum_{a \in A_2} (1 - u_a^\sigma) M \cdot x_a'^* \leq \varepsilon \text{ is true or not. Do}$$

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 $Z_{MINLP,i}^*$ of the MINLP problem. The upper bound of the objective function value is then updated via $\bar{Z}_i = \min\{\bar{Z}_{i-1}, Z_{MINLP,i}^*\}$, whereas the lower bound is updated via $\underline{Z}_i = \max\{\underline{Z}_{i-1}, Z_{R-MINLP,i}^*\}$. 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 $\bar{Z}_i$. Calculate the relative difference between the upper bound and lower bound $|\bar{Z}_i - \underline{Z}_i| / \underline{Z}_i$.

Step 5. Reduce feasible range of variables. Calculate new bounds for each variable $x \in X_{\text{var}}$ 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 $r_x = (\bar{x}^{\text{new}} - \underline{x}^{\text{new}}) / (\bar{x}^{\text{old}} - \underline{x}^{\text{old}})$, $x \in X_{\text{var}}$. If all $r_x \geq r^*$ ($0 < r^* < 1$), 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 χ 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 $\left| \bar{Z}_i - \underline{Z}_i \right| / \underline{Z}_i < \tau$ and the condition $\sum_{a \in A} T_a x_a^\sigma + \sum_{a \in A} T_a R_a p_a^\sigma - \sum_{a \in A} t_a^\sigma x_a'^* - \sum_{a \in A_2} (1 - u_a^\sigma) M \cdot x_a'^* \leq \varepsilon$ 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. (5.14) 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 proposition below, the convergence of the proposed global optimal algorithm is proved.

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

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 $\underline{Z}_i = \max\{\underline{Z}_{i-1}, Z_{R-MILP,i}^*\}$,

$\bar{Z}_i = \min\{\bar{Z}_{i-1}, Z_{MINLP,i}^*\}$ 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 $\underline{Z}_i \leq \hat{Z} \leq Z_{MINLP}^*(\mathbf{y}_i^*, \mathbf{u}_i^*) = \bar{Z}_i$. When the iteration number $i \rightarrow \infty$, the

implementation of Proposition 2 and Proposition 3 in Step 3 guarantees that the

final solution satisfies the inequality

$$\sum_{a \in A} T_a x_{a,i}^* + \sum_{a \in A} T_a R_a p_{a,i}^* - \sum_{a \in A} t_{a,i}^* x_a'^* - \sum_{a \in A_2} (1 - u_{a,i}^*) M \cdot x_a'^* \leq \varepsilon, \quad \text{i.e. the DUE}$$

condition, where ε 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 update the bounds

$\underline{Z}_i$ and $\bar{Z}_i$ for $\hat{Z}$. Therefore, we will have $\lim_{m \rightarrow \infty} \underline{Z}_i = \hat{Z}$ and

$\lim_{m \rightarrow \infty} \bar{Z}_i = \lim_{m \rightarrow \infty} Z_{MINLP}^*(\mathbf{y}_i^*, \mathbf{u}_i^*) = \hat{Z}$. 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.

5.4 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.

5.4.1 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 lead 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:

$$\begin{aligned} x_a &\leq y_a, & \forall a \in A_1 \\ x_a &\leq u_a y_a, & \forall a \in A_2 \end{aligned} \quad (5.85)$$

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 of $k_a = u_a y_a$, $\forall a \in A_2$, constraints (5.85) can be rewritten as linear constraints (5.86) and added to the MINLP and R-MILP model:

$$\begin{aligned} x_a &\leq y_a, & \forall a \in A_1 \\ x_a &\leq k_a, & \forall a \in A_2 \end{aligned} \quad (5.86)$$

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.

5.4.2 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, continuous link capacity variables are assumed. 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 $\{1, 2, \dots, m\}$ represents the feasible number of lanes that contained in a candidate link. Thus domain of y_a can be depicted by

$$y_a \in \{y_a^1, y_a^2, \dots, y_a^m\}, \quad \forall a \in A_2 \quad (5.87)$$

Based on this assumption, for MINLP and R-MINLP, a series of binary variables $\gamma_a^r, r \in \{1, 2, \dots, m\}, a \in A_2$ are introduced 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. (5.40) can be substituted by the following Eq. (5.88):

$$\left. \begin{aligned} y_a &= \sum_{r=1}^m \gamma_a^r y_a^r \\ \sum_{r=1}^m \gamma_a^r &= 1 \\ \gamma_a^r &\in \{0, 1\}, \quad \forall r \in \{1, 2, \dots, m\} \\ \forall a &\in A_2 \end{aligned} \right\} \quad (5.88)$$

Like analysis before, the second and the third constraints in Eq. (5.88) guarantee there is only one γ_a^r can equal to 1 and all the other γ_a^r are forced equal to 0. Thus, from the first constraint, we have $y_a = y_a^r$, only when the associated γ_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. (5.65) and discard y_a from the set Q in Eq. (5.76). The linear relaxation method for the discrete function $\ln(y_a), a \in A_2$ is amended as below:

$$\sum_{r=1}^m \lambda_{ya}^r y_a^r = y_a, \quad y_a^r \in \{y_a^1, y_a^2, \dots, y_a^m\} \quad (5.89)$$

$$\sum_{r=1}^m \lambda_{ya}^r \ln(y_a^r) = L_{ya} \quad (5.90)$$

$$\sum_{r=1}^m \lambda_{ya}^r = 1 \quad (5.91)$$

$$\lambda_{ya}^r \in \{0, 1\}, \quad r = 1, 2, \dots, m \quad (5.92)$$

where only a series of binary variables λ_{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.

5.5 Numerical examples

Figure 5.2 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 Figure 5.2. 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 $t_a(x_a, y_a) = T_a \left(1 + R_a (x_a / y_a)^4 \right)$, where $R_a = 0.15$ and $y_a \in [4, 6]$. The construction cost function $g_a(y_a, u_a) = \alpha u_a y_a + \beta u_a$ is used in the tests. The values of parameters α and β are appropriately set to make the construction cost function value consistent with that in previous studies and given in Table 5.1. Set ε in the VI constraints equals to 5×10^{-5} . The iteration process

terminates if gap between the obtained lower and upper bounds of objective function value is less than 5×10^{-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 the 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 5.1 Value of parameters in construction cost function.

Candidate link	1	2	3	4	5	6
α	1.4200	1.0000	1.9600	2.1200	2.7500	3.0800
β	0.8308	2.1300	2.2691	2.1356	1.3896	3.2568

5.5.1 Example 1: comparison of 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 Luatthep 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 5.3 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 5.2 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 5.3, 1 indicates the corresponding candidate link should be built and 0 otherwise. Objective function value is calculated 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 5.3 and the latter is the final objective function value of the two-step method. In principle, the proposed 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 5.3 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. It is also noted 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 5.3 is that the result with budget 50 is even worse than the result with budget 40 when using the two-step method,

which is 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 the 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 5.4. For each iteration, it presents the evolving upper and lower bounds of objective function value, gap between the two bounds ($\text{Gap} = [\text{Upper bound} - \text{Lower bound}] / \text{Upper bound}$) 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 X_{var} , rescaling the feasible region of variable and other techniques that can improve the efficiency of MILP. This study only sets $X_{\text{var}} = \{\tilde{x}_a, \tilde{h}_a, \forall a \in A; y_a, \forall a \in A_2\}$ and rescales the feasible region of $\tilde{x}_a, \forall a \in A$ 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 the presented 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 5.2 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 5.3 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	Capacity	Objective value	
	New links	Objective value	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 5.4 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

5.5.2 Example 2: DCNDP with discrete levels of capacity improvements

It is also tested that the DCNDP model with discrete levels of capacity improvements. The test network is identical, i.e., the 12-node network shown in Figure 5.2. The assumption is 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 5.5. One can find the optimal objective value is no better than the results of DCNDP as shown in Table 5.3 with continuous capacity enhancement for new link additions, which can be easily interpreted by the more stringent constraint of discrete link capacity.

Table 5.5 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

5.6 Concluding remarks

This study develops 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 and efficiency of the solution algorithm.

Chapter 6 GLOBAL OPTIMIZATION FOR TRANSPORT NETWORK EXPANSION AND SIGNAL SETTING

This chapter proposes a model to address an urban transport planning problem involving combined network design and signal setting in a saturated network. Conventional transport planning models usually deal with the network design problem and signal setting problem separately. However, the fact that network capacity design and capacity allocation determined by network signal setting combines to govern the transport network performance requires the optimal transport planning to consider the two problems simultaneously. In this study, a combined network capacity expansion and signal setting model is developed to consider their mutual interactions so that best transport network performance can be guaranteed. The model is formulated as a bi-level program and a global optimization solution method is designed based on mixed-integer linearization approach to solve the problem, which is inherently non-linear non-convex. Numerical experiments are conducted to demonstrate the model application and the efficiency of solution algorithm.

6.1 Introduction

To solve traffic congestion problem, one important issue in transportation planning is to find out the optimal decision on road capacity design, i.e., the optimal road capacity expansions and optimal planning on new links addition. Another intrinsic issue to be addressed in urban transportation planning is the equilibrium network signal setting (ENSS), which seeks to optimize the road capacity allocation in the network by devising the signal control strategy.

Basically, most of the previous research works investigated the two planning problems, road capacity design in NDP and network signal setting ENSS, separately. However, in practice, NDP determines the link capacity, while ENSS determines the capacity allocation; and it is indeed the combination of the road capacity design and the capacity allocation design that fully determines the resultant equilibrium travel pattern and the transportation network performance. Previous studies addressed the two problems separately, regarding the other problem as exogenously given, which ignores the mutual interaction between them in determining the transport network performance. In this study, the two issues will be unified into one modeling framework as a combined network design and signal setting problem (CNDSSP) to optimize the transportation system performance, which otherwise are considered separately, and may result in urban transportation planning that is way deviated from the true optimal one. Moreover, a global optimization solution method is designed to solve the formulated problem, which is a nonlinear and non-convex model. Following the global optimization solution idea introduced in Wang and Lo (2010), the originally formulated bi-level program is transformed into a single-level mixed-integer linear program by applying a mixed integer linearization approach. Noting that the solution efficiency of this method is highly restricted by the number of introduced integer variables, a new mixed-integer linearization model is applied, which significantly reduces the number of integer variables needed and thus improves the solution efficiency. Specifically, the number of integer variables introduced for the linearization follows logarithmic relationship with the size of the partition scheme, which requires much smaller number of integer variables even if high linearization accuracy and refined partition scheme are needed. In summary, this study contributes to state-of-the-art in two aspects: firstly, a combined network design and signal

setting transport planning model is proposed to take into account their interaction and ensure the true optimal transportation planning. Secondly, an efficient global optimization solution method is developed to seek the globally optimal design of the transportation planning problem. Numerical studies are conducted to demonstrate the solution quality by comparing the proposed model and solution algorithm with the other existing models and solution methods.

The reminder of this chapter is organized as follows. In section 6.2, the bi-level model of combined network design and signal setting is stated and then reformulated as a single-level mixed-integer program (MIP). Section 6.3 demonstrates the single-level model linearization process. A numerical example is given in section 6.4 to illustrate the proposed model and solution method, and to compare the computational results with other models and algorithms. Some final remarks are presented in section 6.5.

6.2 Model formulation

The following notation is used in the formulation.

Parameters:

A^0 the set of links in the network

A the set of links with signal control, $A \subset A^0$

$\tilde{A}$ the set of links without signal control, $\tilde{A} \subset A^0$

A_i the set of approaching links for intersection $i \in I$ in the network

I the set of signal controlled intersections in the network

W the set of origin-destination (OD) pairs

R_w the set of paths between O-D pair $w \in W$

q_w the fixed O-D travel demand, $\forall w \in W$

Δ the link-path incidence matrix, $\Delta = [\delta_{ap}^w]$, $\forall w \in W, a \in A^0, p \in R_w$, where

$\delta_{ap}^w = 1$ if link a belongs to path p between O-D pair w , and $\delta_{ap}^w = 0$ otherwise

ρ the conversion factor which turns travel time into monetary cost

Variables:

x_a the traffic flow of link $a \in A^0$

λ_a the green split of link $a \in A$; $\lambda_a = 1$ if $a \in \tilde{A}$

d_a the queuing delay on link $a \in A$; $d_a = 0$ if $a \in \tilde{A}$

S_a the capacity of link $a \in A^0$ after expansion

f_p^w path flow of path p between O-D pair $w \in W$

c_p^w travel time of path p between O-D pair $w \in W$

π^w minimum path travel cost between O-D pair $w \in W$

Functions:

$t_a(x_a, \lambda_a, S_a)$ travel time function of link $a \in A^0$, which is the function of link flow x_a , green split λ_a and link capacity S_a

$g_a(S_a)$ the link expansion cost function, $\forall a \in A^0$, which is assumed to be the function of S_a

$Q_a(S_a)$ the link storage capacity function, $\forall a \in A^0$, which is assumed to be the function of S_a

6.2.1 Bi-level model for CNDSSP

Given the O-D demand pattern, which is supposed to be relatively stable within the planning horizon and current network layout, the combined continuous network design and signal control problem in this chapter aims to determine which links should be expanded, how much the link capacity should be improved and what is the green split of each signal controlled link. The objective is to minimize the total travel time cost and the investment on link capacity enhancement subject to the feasible range of signal settings and constraints of maximum link capacity expansions.

The combined network design and signal setting problem can be modeled into a bi-level program stated as follows:

$$\text{Minimize } F(\lambda, S, x(\lambda, S), d(\lambda, S)) = \sum_{a \in A^0} \rho x_a \cdot (t_a(x_a, \lambda_a, S_a) + d_a) + \sum_{a \in A^0} g_a(S_a)$$

Subject to

$$\begin{aligned} \lambda_a^{\min} &\leq \lambda_a \leq \lambda_a^{\max}, \quad a \in A \\ \sum_{a \in A_i} \lambda_a &= 1, \quad i \in I \end{aligned} \tag{6.1}$$

$$S_a^0 \leq S_a \leq S_a^{\max}, \quad a \in A^0 \tag{6.2}$$

$$d_a(\lambda, S) \cdot x_a(\lambda, S) \leq Q_a(S_a), \quad a \in A \tag{6.3}$$

While the equilibrium traffic flow x is determined as follows:

$$\text{Minimize } Z(x, f) = \sum_{a \in A^0} \int_0^{x_a} t_a(v, \lambda_a, S_a) dv$$

Subject to

$$\begin{aligned} \sum_{w \in W} \sum_{p \in R_w} f_p^w \delta_{ap}^w &= x_a, \quad a \in A^0 \\ \sum_{p \in R_w} f_p^w &= q^w, \quad \forall w \in W \\ f_p^w &\geq 0, \quad \forall p \in R_w, \quad \forall w \in W \end{aligned} \tag{6.4}$$

$$x_a \leq \lambda_a S_a, \quad a \in A \tag{6.5}$$

Where $\lambda_a^{\max}, \lambda_a^{\min}$ is the upper and lower bound of green split of link $a \in A$;

S_a^0 is the current capacity of link $a \in A$, and $S_a^{\max}$ is the maximum link capacity after expansion due to the restriction of urban land use planning.

In the above formulation, the upper level of the model is to seek the optimal green splits and road capacity enhancement to minimize the total travel time cost and investment on link capacity expansion. Here, total travel time cost includes all flow-dependent running time, signal delay and queuing delay at link exits due to road saturation in the network. As was done in Yang and Yagar (1995), the running time and signal delay are captured by travel time function

$t_a(x_a, \lambda_a)$ and the queuing delay is endogenously solved and denoted as d_a .

Constraint (6.1) states the given green split restriction and entails the conservation equation depending on the specific signal phase structure of an intersection. Eq. (6.2) describes the boundary constraints of link capacity expansion. As a reasonable signal control should consider the queuing length on the approaching leg of an intersection to avoid blocking the upstream section,

Eq. (6.3) enforces that number of queuing vehicles on an approaching leg should be restricted to the link storage capacity. x_a and d_a can be endogenously obtained by solving the lower-level program, which determines the equilibrium travel pattern following the deterministic user equilibrium (UE) principle. Constraint (6.4) ensures that the total travel demand and link flow conservation are fulfilled. Eq. (6.5) denotes link exit capacity constraints. The Lagrange multipliers associated with the capacity constraints are exactly the corresponding queuing delay d_a due to limited link capacity. For any given signal settings and link capacity, drivers' route choice behavior and the resultant equilibrium travel pattern can be predicted through the lower-level program with the assumption that all drivers perfectly know all the travel costs and queuing delay information of the whole network. Note that the queuing delay d_a is an implicit variable in the bi-level model.

6.2.2 Single-level MIP for CNDSSP

This section transforms the bi-level problem into a single-level program, which is then linearized into a mixed-integer linear program (MILP), whose global optimality can be guaranteed.

The UE condition assumes that each driver has perfect information of travel time on each possible path and chooses the route with minimum travel time, i.e., the travel times on all the routes used are equal and less than those experienced on any unused routes (Wardrop, 1952). It implies that the following relationship should be satisfied:

$$\begin{cases} f_p^w \geq 0 \Rightarrow c_p^w - \pi^w = 0 \\ f_p^w = 0 \Rightarrow c_p^w - \pi^w \geq 0 \end{cases}, \quad \forall p \in R_w, \quad w \in W \quad (6.6)$$

Wherein $c_p^w = \sum_{a \in A^0} \delta_{ap}^w (t_a(x_a, \lambda_a, S_a) + d_a)$, which is the total travel time cost for individual traveler, including both d_a the queuing delay and t_a the running time and signal delay.

The above UE condition can be expressed into the form of complementary condition:

$$\begin{cases} f_p^w \cdot (c_p^w - \pi^w) = 0 \\ c_p^w - \pi^w \geq 0 \end{cases}, \quad \forall p \in R_w, \quad w \in W \quad (6.7)$$

Next, a set of mixed-integer constraints can be introduced to substitute Eqs. (6.7). Referring to Lo (1999; 2001) and Wang and Lo (2010), the equivalent set of mixed-integer constraints are stated as follows:

$$\begin{cases} L \cdot \sigma_p^w + \varepsilon \leq f_p^w \leq U \cdot (1 - \sigma_p^w) \\ L \cdot \sigma_p^w \leq c_p^w - \pi^w \leq U \cdot \sigma_p^w \\ c_p^w - \pi^w \geq 0 \\ \sigma_p^w \in \{0, 1\} \\ \forall p \in R_w, \quad w \in W \end{cases} \quad (6.8)$$

Where L is a large enough negative number; U is a large enough positive number; ε is a very small positive number; σ_p^w is a 0-1 binary variable. If $\sigma_p^w = 0$, then $f_p^w > 0$; on the contrary, if $\sigma_p^w = 1$, f_p^w equal to zero. The detailed verification of the equivalence between the set of constraints (6.8) and UE condition can be referred to Lo (1999; 2001) and Wang and Lo (2010). Note that constraints (6.8) are now linear in f_p^w , σ_p^w , c_p^w and π^w .

In the bi-level formulation, queuing delay d_a is implicit as mentioned above. To transform the bi-level program into single level problem, d_a should be

converted into an explicit variable. The following process shows how d_a is converted and how to formulate the mixed-integer constraints associated with Eq. (6.5).

Since queues may only occur on saturated roads, the following relationship should be enforced (Yang and Yagar, 1995):

$$\begin{cases} \lambda_a S_a \geq x_a \Rightarrow d_a = 0 \\ \lambda_a S_a = x_a \Rightarrow d_a \geq 0 \end{cases}, \quad \forall a \in A \quad (6.9)$$

Which means if a link is unsaturated, i.e., $\lambda_a S_a \geq x_a$, then the queuing delay on this link is zero; otherwise, if a link is over-saturated, the queuing delay is greater than zero. The constraint (6.9) is similar to that in (6.6). Therefore, constraint (6.9) can also be rewritten into the form of complementary problem in (6.10):

$$\begin{cases} (\lambda_a S_a - x_a) \cdot d_a = 0 \\ \lambda_a S_a - x_a \geq 0 \end{cases}, \quad \forall a \in A \quad (6.10)$$

Then following the transformation procedure as imposed on (6.7), the constraints in terms of mixed-integer linear constraints in (6.11) can be developed to replace (6.10):

$$\begin{cases} L \cdot \gamma_a + \varepsilon \leq \lambda_a S_a - x_a \leq U \cdot (1 - \gamma_a) \\ L \cdot \gamma_a \leq d_a \leq U \cdot \gamma_a \\ \lambda_a S_a - x_a \geq 0 \\ \gamma_a \in \{0, 1\} \\ \forall a \in A \end{cases} \quad (6.11)$$

Where γ_a is a binary variable. For $\gamma_a = 0$, we have:

$$\gamma_a = 0 \Rightarrow \begin{cases} \varepsilon \leq \lambda_a S_a - x_a \leq U \Leftrightarrow \lambda_a S_a - x_a > 0 \\ 0 \leq d_a \leq 0 \Leftrightarrow d_a = 0 \end{cases} \quad (6.12)$$

On the other hand, if $\sigma_p^w = 1$, since $d_a \geq 0$, we have:

$$\gamma_a = 1 \Rightarrow \begin{cases} \lambda_a S_a - x_a \geq 0 \\ L + \varepsilon \leq \lambda_a S_a - x_a \leq 0 \Leftrightarrow \lambda_a S_a - x_a \leq 0 \\ L \leq d_a \leq U \\ d_a \geq 0 \end{cases} \Leftrightarrow \begin{cases} \lambda_a S_a - x_a = 0 \\ d_a \geq 0 \end{cases} \quad (6.13)$$

Thus, the set of constraints (6.11) is equivalent to the “if-then” conditions in (6.9). Note that d_a is an explicit variable in the constraints (6.11), which are linear in x_a , d_a and γ_a , but remain non-linear in S_a and λ_a .

In summary, the single-level MIP for CNDSSP can be formulated as follows:

$$\text{Min } F(\lambda, S, x(\lambda, S), d(\lambda, S)) = \sum_{a \in A^0} \rho x_a \cdot (t_a(x_a, \lambda_a, S_a) + d_a) + \sum_{a \in A^0} g_a(S_a) \quad (6.14)$$

Subject to:

(i) Signal variable constraints:

$$\begin{aligned} \lambda_a^{\min} &\leq \lambda_a \leq \lambda_a^{\max}, \quad a \in A \\ \sum_{a \in A_i} \lambda_a &= 1, \quad i \in I \end{aligned} \quad (6.15)$$

(ii) Capacity constraints and demand conservation constraints:

$$\begin{aligned} S_a^0 &\leq S_a \leq S_a^{\max}, \quad a \in A^0 \\ d_a \cdot x_a &\leq Q_a(S_a), \quad \forall a \in A \\ q^w &= \sum_{p \in R_w} f_p^w, \quad \forall w \in W \end{aligned} \quad (6.16)$$

(iii) UE constraints:

$$\begin{cases}
L \cdot \sigma_p^w + \varepsilon \leq f_p^w \leq U \cdot (1 - \sigma_p^w) \\
L \cdot \sigma_p^w \leq c_p^w - \pi^w \leq U \cdot \sigma_p^w \\
c_p^w - \pi^w \geq 0 \\
\sigma_p^w \in \{0, 1\} \\
\forall p \in R_w, \quad w \in W
\end{cases} \quad (6.17)$$

(iv) Queuing delay constraints:

$$\begin{cases}
L \cdot \gamma_a + \varepsilon \leq \lambda_a S_a - x_a \leq U \cdot (1 - \gamma_a) \\
L \cdot \gamma_a \leq d_a \leq U \cdot \gamma_a \\
\lambda_a S_a - x_a \geq 0 \\
\gamma_a \in \{0, 1\} \\
\forall a \in A
\end{cases} \quad (6.18)$$

(v) Definitional constraints:

$$\begin{aligned}
x_a &= \sum_{w \in W} \sum_{p \in R_w} \delta_{ap}^w \cdot f_p^w, \quad \forall a \in A^0 \\
c_p^w &= \sum_{a \in A} \delta_{ap}^w \cdot (t_a(x_a, \lambda_a, S_a) + d_a), \quad \forall p \in R_w, \quad w \in W \\
x_a &\geq 0, \quad \forall a \in A^0 \\
d_a &\geq 0, \quad \forall a \in A \\
f_p^w &\geq 0, \quad \forall p \in R_w, \quad w \in W
\end{aligned} \quad (6.19)$$

In the above single-level mixed-integer model for CNDSSP, the UE conditions and queuing delay constraints are represented by a set of mixed-integer constraints (6.17)-(6.18), rather than in a lower-level program formulation. One can note that the nonlinearity stems from the link travel time function, the objective function and capacity and queuing delay constraints, i.e., this single-level model is a mixed-integer non-linear program (MINLP). In the next section, a linearization technique will be employed to transform the MINLP into a MILP, through which global optimal solution can be obtained.

6.3 Linearization of the single-level model

Mixed integer models would be used to formulate the nonlinear terms into piecewise linear approximations. Indeed, many mixed-integer models have been studied to capture piecewise linear functions (Croxton et al., 2003; Keha et al., 2004; Lee and Wilson, 2001; Sherali, 2001). However, different formulations have different properties and their computational efficiencies vary from each other. Recently, several new and existing modeling efforts of piecewise-linear functions are reviewed in Vielma et al. (2010). This study applies the logarithmic formulation Log-model (Logarithmic branching convex combination model) to linearize the nonlinear terms, which significantly reduces the number of variables and constraints needed by introducing a binary vector in $\{0,1\}^{\lceil \log_2 |P| \rceil}$ through an injective function $P \rightarrow \{0,1\}^{\lceil \log_2 |P| \rceil}$ to identify an active polytope P in the linearization process. Specifically, the logarithmic relationship between the number of binary variables and the number of polytopes and constraints makes sure that the size of the linearization model will not be prohibitively large even when high approximation accuracy and refined partition scheme are required.

This section includes three parts. The details of linearizing the two-dimensional functions and MILP formulation are expressed in 6.3.1. The objective function and the storage capacity function are then transformed in 6.3.2. The MILP model for the CNDSSP is summarized in 6.3.3. Finally, the MILP formulation is rectified for the case of discrete link capacity improvement.

6.3.1 Linearization of two-dimensional functions

In the single-level MIP proposed above, there are several non-linear terms (i.e., $d_a \cdot x_a$, $\lambda_a \cdot S_a$ and $t_a(x_a, \lambda_a, S_a)$) and all of them are multivariate functions

with multiple variables. In this study, the nonlinear terms will be approximated by piecewise linear functions. In this way, the MINLP can be transformed into the MILP.

For the last non-linear term $t_a(x_a, \lambda_a, S_a)$, it is used to cover both running time and signal delay at link exit. The signal delay stems from the signal interruption of traffic and could be determined by approximated formula developed from local traffic conditions, such as Webster's two-term delay formula for signalized intersections (Webster, 1958) and the polynomial cost function (Van Vuren and Van Vliet, 1992, Chapter 5). However, as was pointed out in Yang and Yagar (1995), both of the two formulas have drawbacks in describing the signal delay, especially for the saturated links. Therefore, following Yang and Yagar (1995), the link performance function is used to depict both the running time on the link and signal delay at link exit. The formulation is given as follows:

For signal controlled link $a \in A$:

$$t_a(x_a, \lambda_a, S_a) = t_a^0 [1 + (\frac{x_a}{\lambda_a S_a})^2] \quad (6.20)$$

For non-signal controlled link $a \in \tilde{A}$:

$$t_a(x_a, S_a) = t_a^0 [1 + (\frac{x_a}{S_a})^2] \quad (6.21)$$

Where t_a^0 is free flow travel time of link a and given as known constant. Note that equation (6.20) is a function with three variables and it includes the term $\lambda_a \cdot S_a$ in its formulation. If the term $\lambda_a \cdot S_a$ is linearized first and substituted by a new variable, Eq. (6.20) will have the same form as Eq. (6.21) and both of

them are two-dimensional. Since the other two non-linear terms, i.e., $\lambda_a \cdot S_a$ and $d_a \cdot x_a$, are both two-dimensional, all the non-linear terms in the single-level model can be regarded as two-dimensional and be linearized through the same piecewise-linear approximation method. Now $\lambda_a \cdot S_a$ is taken as the example to demonstrate the detailed piecewise-linearization process by applying Log model.

Let $\lambda_a \cdot S_a$ be represented by a new variable B_a , i.e., $B_a = B(S_a, \lambda_a) = \lambda_a \cdot S_a$. The bounded domain of B_a would be portioned into a finite family of polytopes P , which is usually taken to be a triangulation of the field (Lee and Wilson, 2001). For Log model, the family of polytopes needs to be topologically compatible or equivalent with a triangulation known as J_1 or “Union Jack” (Todd, 1977). The Log model needs only logarithmic number of binary variables and thus a branching scheme with a logarithmic number of dichotomies, which is introduced in Vielma and Nemhauser (2011). In this chapter, the domain of B_a is be divided into subsets as shown in Figure 6.1 when $P = J_1$. The feasible region of S_a (i.e., $[S_a^0, S_a^{\max}]$) is partitioned into $2N$ intervals in x dimension, i.e., $[S_a^{n-1}, S_a^n]$, $n=1, 2, \dots, 2N$, and λ_a (i.e., $[\lambda_a^{\min}, \lambda_a^{\max}]$) is partitioned into $2M$ intervals in y dimension, i.e., $[\lambda_a^{m-1}, \lambda_a^m]$, $m=1, 2, \dots, 2M$. Note that the lengths of these intervals are not necessarily the same.

Apparently, with the increasing number of N and M , the piecewise-linear approximation will be more accurate. However, the approximation accuracy does not only depend on the total number of polytopes; the detailed partition scheme also impacts the results remarkably. For example, in Figure 6.2,

although the piecewise linear function L_1 only has two pieces, the minimum of L_1 is equal to the actual minimum of original function. As compared to L_1 , the minimum of L_2 , which has three segments, is larger than the actual minimum. Hence, advanced partition method, which better describes the profile characteristics of the graph of non-linear functions, can be studied in the future. In this chapter, the commonly used evenly-partitioned scheme is applied.

Figure 6.1 An equivalent J_1 triangulation of the feasible region

Following Log method in Vielma et al. (2010), the function $B_a = B(S_a, \lambda_a) = \lambda_a \cdot S_a$ for $a \in A$ can be substituted by a set of MILP constraints as follows:

$$\begin{aligned} \sum_{v \in V(P)} \alpha_v (v_x, v_y) &= (S_a, \lambda_a) \\ \sum_{v \in V(P)} \alpha_v B(v_x, v_y) &= B_a \end{aligned} \quad (6.22)$$

$$\begin{aligned} \alpha_v &\geq 0 \quad \forall v \in V(P) \\ \sum_{v \in V(P)} \alpha_v &= 1 \end{aligned} \quad (6.23)$$

$$\begin{aligned}
\sum_{v \in L_k} \alpha_v &\leq y_k \\
\sum_{v \in R_k} \alpha_v &\leq 1 - y_k \\
y_k &\in \{0,1\} \quad \forall k \in K
\end{aligned} \tag{6.24}$$

Figure 6.2 Different piecewise-linear approximations for $f(x)$

In the above formulation, P is the family of partitioned triangles and $v \in V(P)$ denotes corner-point, which may belong to more than one triangle in the partitioned domain; (v_x, v_y) represents the coordinates of corner-point v ; α_v is the weighted factor of convex combination related to corner-point v , and hence, $\alpha_v \in [0,1]$. Since B_a must fall into a single triangle of the domain, i.e. the active triangle, B_a can be approximated in the form of a weighted function of $B(v_x, v_y)$ value of vertices of this triangle; (S_a, λ_a) can also be calculated from a weighted function of the coordinates of the vertices of the active triangle, as is shown in Eqs. (6.22). Eqs. (6.23) entail that the sum of weighted factors α_v of all vertices in the domain is equal to 1. Indeed, only α_v of the active

triangle is larger than zero while the weighted factors of the other inactive triangles are all equal to zero.

As α_v is not associated with any specific polytope, constraints (6.22)-(6.23) only require the non-negativity of α_v . However, constraints (6.24) ensure that all nonzero α_v are related to the corner-points of one polytope in P . In (6.24),

$L_k, R_k \subset V(P)$ and the set of dichotomies $\{L_k, R_k\}_{k \in K}$ is referred to a branching scheme for the family of polytopes P that is equivalent to J_1 . For each particular triangle $P^* \in P$, the vertices of P^* can be expressed by $V(P^*) = \bigcap_{k \in K} (V(P) \setminus T_k)$, where $T_k = L_k$ or $T_k = R_k$ for each $k \in K$. The index set K is divided into two subsets K_1 and K_2 . The first is given by

$K_1 = \{(kx, ky) \mid kx \in \{1, \dots, Dim\}, ky \in \{1, \dots, \lceil \log_2 N_{kx} \rceil\}\}$, where Dim refers to dimension number of the domain and it is equal to two in this case because the non-linear function B_a is two-dimensional; N_{kx} is the number of intervals divided in kx dimension and note that N_{kx} is even because of J_1 triangulation.

Assume each interval index $n \in \{1, 2, \dots, N_{kx}\}$ can be represented by a fixed binary vector $G^n \in \{0, 1\}^{\lceil \log_2 N_{kx} \rceil}$. The selection of G^n is arbitrary on condition that the two successive vectors G^n and G^{n+1} differ in only one bit for each $n \in \{1, 2, \dots, N_{kx} - 1\}$. The set of binary vectors with this property is known as Gray code or reflected binary code (Breckman, 1956). Then for each

$$(kx, ky) \in K_1, \quad L_{(kx, ky)} = \{v \in V \mid v_{kx} \in O(ky, 1)\} \quad \text{and}$$

$$R_{(kx, ky)} = \{v \in V \mid v_{kx} \in O(ky, 0)\}, \quad \text{where}$$

$$O(l, b) = \{n \in \{0, 1, \dots, N_{kx}\} \mid (n = 0 \text{ or } G_l^n = b) \text{ and } (n = N_{kx} \text{ or } G_l^{n+1} = b)\} \quad . \quad \text{The}$$

second set K_2 is denoted by $K_2 = \{(kx, ky) \mid kx, ky \in \{1, \dots, Dim\} \text{ and } kx < ky\}$.

For each $(kx, ky) \in K_2$, $L_{(kx, ky)} = \{v \in V \mid v_{kx} \text{ is even and } v_{ky} \text{ is odd}\}$ and $R_{(kx, ky)} = \{v \in V \mid v_{kx} \text{ is odd and } v_{ky} \text{ is even}\}$. The size of Eqs. (6.24) is equal to $2[Dim \cdot \lceil \log_2 N_{kx} \rceil + Dim(Dim-1)/2]$ and only dependent on $\lceil \log_2 N_{kx} \rceil$ when Dim is fixed. A simple example of this branching scheme of an equivalent J_1 triangulation is given in details in Appendix E.

Similarly, a new variable H_a is introduced to denote the term of $d_a \cdot x_a$ in the single-level problem. Then, Log model is applied for piecewise-linear approximation, and the function $H_a = H(d_a, x_a) = d_a \cdot x_a$ for $a \in A$ can be written into a set of MILP constraints as follows:

$$\begin{aligned} \sum_{v \in V(P_1)} \beta_v(v_x, v_y) &= (d_a, x_a) \\ \sum_{v \in V(P_1)} \beta_v H(v_x, v_y) &= H_a \end{aligned} \quad (6.25)$$

$$\begin{aligned} \beta_v &\geq 0 \quad \forall v \in V(P_1) \\ \sum_{v \in V(P_1)} \beta_v &= 1 \end{aligned} \quad (6.26)$$

$$\begin{aligned} \sum_{v \in L_k} \beta_v &\leq z_k \\ \sum_{v \in R_k} \beta_v &\leq 1 - z_k \\ z_k &\in \{0, 1\} \quad \forall k \in K \end{aligned} \quad (6.27)$$

Where $P_1 = J_1$ is the family of polytopes, which are partitioned triangles of the domain of H_a , i.e. $D_H = \{(d_a, x_a) \mid d_a \in [d_a^{\min}, d_a^{\max}], x_a \in [x_a^{\min}, x_a^{\max}]\}$. For link flow x_a , the upper bound $x_a^{\max}$ can be set as the sum of travel demands of the O-D pairs, at least one viable path of which includes link a , or simply the sum of all O-D demands. Since O-D demands are given as constants in the proposed

model, the upper bound $x_a^{\max}$ can be easily defined. As for queuing delay d_a , there is no explicit upper bound in the constraints. However, considering the realistic condition of queuing delay at an intersection, d_a should not be unbounded. $d_a^{\max}$ can be set as a positive large enough number which fully covers the feasible region of d_a , e.g. four times signal cycle or free flow travel time of the whole link a . Both $x_a^{\min}$ and $d_a^{\min}$ are set equal to zero.

Now consider the linearization of link travel time function. For signal controlled link, $\lambda_a \cdot S_a$ is substituted by B_a . Thus, the link travel time function can be regarded as a function with respect to both x_a and B_a , i.e., $t_a(x_a, B_a) = t_a^0 [1 + (x_a / B_a)^2]$, $a \in A^0$, the piecewise-linear approximation of which is represented as follows:

$$\begin{aligned} \sum_{v \in V(P_2)} \gamma_v (v_x, v_y) &= (x_a, B_a) \\ \sum_{v \in V(P_2)} \gamma_v t_a(v_x, v_y) &= t_a \end{aligned} \quad (6.28)$$

$$\begin{aligned} \gamma_v &\geq 0 \quad \forall v \in V(P_2) \\ \sum_{v \in V(P_2)} \gamma_v &= 1 \end{aligned} \quad (6.29)$$

$$\begin{aligned} \sum_{v \in L_k} \gamma_v &\leq u_k \\ \sum_{v \in R_k} \gamma_v &\leq 1 - u_k \\ u_k &\in \{0, 1\} \quad \forall k \in K \end{aligned} \quad (6.30)$$

Where $P_2 = J_1$ is the family of polytopes which is the triangulation of t_a . The range of B_a is defined as $[B_a^{\min}, B_a^{\max}]$, where $B_a^{\min} = \lambda_a^{\min} \cdot S_a^0$ and $B_a^{\max} = \lambda_a^{\max} \cdot S_a^{\max}$. For non-signal controlled link $a \in \tilde{A}$, since

$t_a(x_a, S_a) = t_a^0[1 + (x_a / S_a)^2]$ is originally two-dimensional, the piecewise-linear functions can still be expressed as (6.28)-(6.30) by changing B_a to S_a .

6.3.2 Linearization of objective function and storage capacity function

The objective function of the single-level optimization model is the combination of actual total system travel cost and associated link expansion cost. The first term of the objective, i.e., $\sum_{a \in A^0} \rho x_a \cdot (t_a(x_a, \lambda_a, S_a) + d_a)$, is non-

linear and non-convex. But through the UE principle, it is easy to rewrite the first part into a linear form. When UE condition is satisfied, all used paths of the same O-D pair must equal to the minimum travel time π^w , which includes running time, signal delay and queuing delay in this model. Then, the total system travel cost can be rewritten into the equivalent form of $\sum_{w \in W} \rho q^w \pi^w$,

where all O-D demands q^w are given constants and ρ is the factor to convert travel time to monetary travel cost, which is also a constant. Thus, the first term of the objective function is linear. The second term of the objective function $\sum_{a \in A^0} g_a(S_a)$, which determines the link expansion expense, is a function of S_a .

In practical use, due to various topographic and geological conditions of land, the link investment cost function can be quite different. If $g_a(S_a)$ is linear, the whole objective function is linear, and no transformation is needed. Otherwise, the linearization method Log applied in the last section can be used to transform the non-linear functions into piecewise linear approximations. As $g_a(S_a)$ is only a single variable function, the linearization process is much easier than multivariate functions, like $t_a(x_a, \lambda_a, S_a)$. The domain of g_a is one-dimensional and partitioned into line segments rather than triangles. In

conclusion, the investment function can also be expressed in a linear form and thus the whole objective function is linear.

For the link storage capacity function $Q_a(S_a)$, which is also a single variable function with respect to S_a , it can be treated in the identical way as in the link expansion cost function. That is, a linear or piecewise-linearized link storage capacity function Q_a can always be acquired. In this chapter, for illustration purpose, linear functions of $g_a(S_a)$ and $Q_a(S_a)$ is simply assumed.

6.3.3 The reformulated MILP model

In summary, the reformulated MILP model of the original single-level optimization program is summarized as follows:

$$\text{Minimize } F(\lambda, S, x, d) = \sum_{w \in W} \rho q^w \pi^w + \sum_{a \in A^0} g_a(S_a) \quad (6.31)$$

Subject to:

(i) Green split constraints for links with signal control:

$$\begin{aligned} \lambda_a^{\min} &\leq \lambda_a \leq \lambda_a^{\max}, \quad a \in A \\ \sum_{a \in A_i} \lambda_a &= 1, \quad i \in I \end{aligned} \quad (6.32)$$

(ii) Capacity constraints and demand conservation constraints:

$$\begin{aligned} S_a^0 &\leq S_a \leq S_a^{\max}, \quad a \in A^0 \\ H_a &\leq Q_a(S_a), \quad \forall a \in A \\ q^w &= \sum_{p \in R_w} f_p^w, \quad \forall w \in W \end{aligned} \quad (6.33)$$

(iii) UE condition constraints:

$$\begin{cases}
L \cdot \sigma_p^w + \varepsilon \leq f_p^w \leq U \cdot (1 - \sigma_p^w) \\
L \cdot \sigma_p^w \leq c_p^w - \pi^w \leq U \cdot \sigma_p^w \\
c_p^w - \pi^w \geq 0 \\
\sigma_p^w \in \{0, 1\} \\
\forall p \in R_w, \quad w \in W
\end{cases} \quad (6.34)$$

(iv) Queuing delay constraint :

$$\begin{cases}
L \cdot \gamma_a + \varepsilon \leq B_a - x_a \leq U \cdot (1 - \gamma_a) \\
L \cdot \gamma_a \leq d_a \leq U \cdot \gamma_a \\
x_a \leq B_a \\
\gamma_a \in \{0, 1\} \\
\forall a \in A
\end{cases} \quad (6.35)$$

(v) Linearization of function $B_a = \lambda_a S_a$ for $a \in A$:

$$\begin{cases}
\sum_{v \in V(P)} \alpha_v(v_x, v_y) = (S_a, \lambda_a) \\
\sum_{v \in V(P)} \alpha_v B(v_x, v_y) = B_a \\
\alpha_v \geq 0 \quad \forall v \in V(P) \\
\sum_{v \in V(P)} \alpha_v = 1 \\
\sum_{v \in L_k} \alpha_v \leq y_k \\
\sum_{v \in R_k} \alpha_v \leq 1 - y_k \\
y_k \in \{0, 1\} \quad \forall k \in K
\end{cases} \quad (6.36)$$

(vi) Linearization of function $H_a = d_a x_a$ for $a \in A$:

$$\left\{ \begin{array}{l} \sum_{v \in V(P_1)} \beta_v(v_x, v_y) = (d_a, x_a) \\ \sum_{v \in V(P_1)} \beta_v H(v_x, v_y) = H_a \\ \beta_v \geq 0 \quad \forall v \in V(P_1) \\ \sum_{v \in V(P_1)} \beta_v = 1 \\ \sum_{v \in L_k} \beta_v \leq z_k \\ \sum_{v \in R_k} \beta_v \leq 1 - z_k \\ z_k \in \{0, 1\} \quad \forall k \in K \end{array} \right. \quad (6.37)$$

For links with signal control $a \in A$, linearization of the link travel time function t_a is as follows; while for links without signal control $a \in \tilde{A}$, B_a in the following constraints is substituted by S_a .

$$\left\{ \begin{array}{l} \sum_{v \in V(P_2)} \gamma_v(v_x, v_y) = (x_a, B_a) \\ \sum_{v \in V(P_2)} \gamma_v t_a(v_x, v_y) = t_a \\ \gamma_v \geq 0 \quad \forall v \in V(P_2) \\ \sum_{v \in V(P_2)} \gamma_v = 1 \\ \sum_{v \in L_k} \gamma_v \leq u_k \\ \sum_{v \in R_k} \gamma_v \leq 1 - u_k \\ u_k \in \{0, 1\} \quad \forall k \in K \end{array} \right. \quad (6.38)$$

(vii) Demand conservation and definitional constraints:

$$\begin{aligned}
x_a &= \sum_{w \in W} \sum_{p \in R_w} \delta_{ap}^w \cdot f_p^w, \quad \forall a \in A^0 \\
c_p^w &= \sum_{a \in A^0} \delta_{ap}^w \cdot (t_a(x_a, \lambda_a, S_a) + d_a), \quad \forall p \in R_w, \quad w \in W \\
x_a &\geq 0, \quad \forall a \in A^0 \\
d_a &\geq 0, \quad \forall a \in A \\
f_p^w &\geq 0, \quad \forall p \in R_w, \quad w \in W
\end{aligned} \tag{6.39}$$

So far, the CNDSSP model with queuing in saturated network is cast into a mixed MILP. It can thus be easily solved by existing commercial MILP solvers like CPLEX, and the solution possesses global optimization property.

6.3.4 Adapting the MILP for discrete link capacity improvement

In reality, the expansion of roadway segments may be on discrete levels, which means the road capacity expansion is determined by the number of lanes added. To deal with the CNDSSP on discrete levels of link capacity improvement, some amendments will be made on the above MILP model to adapt it for solving this problem.

With discrete link capacity improvement, the CNDSSP problem changes to decide how many lanes should be added on each link in addition to set signal timings, which is a mixed discrete and continuous problem. Let Ln_a denote the number of lanes to be added to link $a \in A^0$ and LN_a be the maximum number of lanes that can be added to link a . Supposing the capacity raised by each additional lane of link a is represented by a constant Lc_a . Thus, the discrete variable of link capacity can be written as:

$$S_a = S_a^0 + Ln_a \cdot Lc_a, \quad \forall a \in A^0 \tag{6.40}$$

wherein S_a^0 is the original capacity of link a , $Ln_a \in \{1, 2 \dots LN_a\}$. Hence, the restriction of road capacity is decided by the maximum number of lanes LN_a added to that link and then the first constraint of (6.33) can be removed. Now, it only need further express the integer variable Ln_a into a combination of a series of binary variables, as is shown below:

$$\begin{cases} Ln_a = \sum_{i=1}^{LN_a} i \cdot \zeta_a^i \\ \sum_{i=1}^{LN_a} \zeta_a^i = 1 \\ \zeta_a^i \in \{0, 1\}, \forall i \in \{1, 2 \dots LN_a\} \\ \forall a \in A^0 \end{cases} \quad (6.41)$$

Where ζ_a^i is the binary variable that represents adding a number of i lanes to link a . Constraints (6.41) state that if a number of j lanes are added, i.e. $Ln_a = j$, then $\zeta_a^j = 1$ and all the other binary variables $\zeta_a^i, \forall i = 1, 2 \dots LN_a$ and $i \neq j$, are equal to zero. Thus, the number of additional lanes Ln_a is determined by the set of ζ_a^i . As the constraints (6.40)-(6.41) are linear, they can be directly attached to the MILP (6.31)-(6.39) and then be solved by commercial software. In conclusion, through the adaptation made above, the MILP for CNDSSP with continuous road capacity improvement can also be extended to solve the CNDSSP with discrete link capacity.

6.4 Numerical studies

In the numerical studies, the performance of the proposed CNDSSP model and the Log-MILP method is tested by two examples in different networks. The results of other models and local optimal method are also be compared. The

Log-MILP solution method of the two cases is modeled by YALMIP-R20120420 (Löfberg, 2004) and then solved by external commercial optimization solver CPLEX optimization studio 12.3 (IBM ILOG, 2009). All experiments are taken on a personal computer with Intel(R) Core(TM) i7 860 @ 2.80GHz CPU, 8GB RAM and Windows 7 Enterprise operating system (64-bit).

6.4.1 Case 1: the Yang-Yagar network

Figure 6.3 The small Yang-Yagar network.

This case study follows the numerical example used in Yang and Yagar (1995). Through this example, the CNDSSP model is compared with two other models: ENSS and CNDP. The global optimal solution from the Log-MILP method is also compared with result from the method of sensitivity analysis, which is indeed local optimal. The small test network with one signal controlled intersection (node 3) is depicted in Figure 6.3. Link parameters including free-flow travel time, expansion cost factor, current capacity and expansion bound are listed in Table 6.1. O-D demands, bound of green splits and link storage capacity multiplier are all shown in Table 6.2.

Table 6.1 Link data input for the small network.

Link	Free-flow time	Expansion cost factor	Current capacity	Bound of capacity expansions
1	4	2	60	80
2	5	3	50	80
3	2	5	50	70
4	8	4	50	60
5	8	1	80	100
6	4	3	60	90
7	4	6	50	70
8	2	2	60	80
9	3	4	50	70

Table 6.2 Other parameters used in the network.

Variable		Value
Demand	1→5	70
	2→4	80
Bound of green split	$\lambda_a^{\min}$	0.05
	$\lambda_a^{\max}$	0.95
Maximum storage capacity multiplier:	k_1	1.7
$Q_a(S_a) = k_a \cdot S_a$	k_2	2.0

6.4.1.1 Comparison of CNDSSP model with other models

In solving the proposed CNDSSP model by using the Log-MILP algorithm, since feasible regions of x_a , s_a and B_a are much larger than those of λ_a and d_a , the number of partitioned segments of x_a , s_a and B_a should also be larger to ensure the accuracy as well as the computation speed at the same time. Here, the domains for x_a , s_a and B_a are all separated into 80 equal segments, whereas those of λ_a and d_a are both separated into 20 equal segments. This partition scheme is adopted throughout the tests with the solution method of Log-MILP in subsection 6.4.1. Partition scheme has great effects on the accuracy of solutions and the computational time. If a finer solution is needed, a much more refined partition scheme may be selected. It is obvious that the

bounds of the variables should be tight to avoid excessive branching and superfluous computational time. The upper bound of x_a is set as the sum of the demands of O-D pairs, at least one path of which involves the link. Assume the upper bound of d_a is 10 which is even larger than the maximum link free flow travel time.

Table 6.3 displays the optimal solution and objective value of CNDDSP solved by the proposed Log-MILP method. The optimal solution involves link improvement and signal settings (suppose the original green ratio of link 1 is 0.5) and F is the objective value calculated directly from the Log-MILP. Table 6.3 shows the exact objective function values recalculated from the UE traffic assignment with side constrained link capacity (Larsson and Patriksson, 1995) by substituting the optimal solution of link capacity and signal setting from the Log-MILP, which are indicated by F_r . In order to compare the proposed CNDDSP model with the traditional CNDP and the ENSS model, the test network is improved by applying the three models individually. All these models are solved by the Log-MILP algorithm. Table 6.3 also reports the final results obtained from these models. Before any model is applied on the test network, i.e. the case labeled ‘do nothing’ in Table 6.3, the value of the objective function is 2415.2 and the queuing delay on link 1 and link 2 are 1.7439 and 2.3188 respectively (Yang and Yagar, 1995). When the ENSS model is imposed, the green split of link 1 is shifted to 0.2333, which reduces the objective value to 2398.8. This model improves the performance of the network by 0.68% ($= [2398.8 - 2415.2] / 2415.2$). However, the queuing delay on both two approaching legs (i.e. link 1 and link 2) still exists and the delay on link 1 even increases. When using the CNDP model, the optimal solution involves the capacity enhancement on link 1, 2, 5 and 8, supposing the green

ratio of link 1 is fixed. The value of objective function decreases greatly to 2177.1 and the performance of the network increases by 9.86% ($= [2177.1 - 2415.2] / 2415.2$). Besides, the queuing delay on both link 1 and link 2 reduces to zero. When the proposed CNDSSP model is applied on the network, the objective value is the best among the results of the three models, which decreases by 10.20% ($= [2168.8 - 2415.2] / 2415.2$) from the do-nothing case, and there is also no queuing delay on both two links. The gaps between the objective values directly from the proposed Log-MILP method and the exact ones are no larger than 0.0092% (from the CNDSSP model, $= [2168.8 - 2168.6] / 2168.6$), which are quite small. This may be due to the small size of the test network. The computational time for the three models is 0.21 min (ENSS), 1.46 min (CNDP) and 7.75 min (CNDSSP), respectively.

Table 6.3 Comparison of CNDSSP with other models.

Variable	Do nothing	ENSS	CNDP	CNDSSP
Green split of link 1	0.5	0.2333	0.5	0.3985
Capacity enhancement				
Δs_1	-	-	13.2500	13.9359
Δs_2	-	-	26.2500	24.0962
Δs_5	-	-	20	20
Δs_8	-	-	0.0873	
Queuing delay				
d_1	1.7439	3.3883		
d_2	2.3188	0.6030		
Objective value				
F	-	2398.8	2177.1	2168.8
F_r	2415.2	2398.7	2177.0	2168.6

6.4.1.2 Comparison of solution results from Log-MILP method with sensitivity analysis approach

It is hard to test the performance of the Log-MILP method through the CNDSSP model because there is no benchmark algorithm to solve this problem in the literature. However, the ENSS problem can be seen as a special case of CNDSSP, where the link capacity is fixed and only the green split is variable. Since the ENSS problem has been tested with sensitivity analysis solution approach in Yang and Yagar (1995), it is adopted to be compared with the presented global optimization algorithm. The former is a typical local heuristic algorithm. It is efficient and widely implemented in solving bi-level optimization problem, though only local optimal points may be found due to non-linear non-convex character of the bi-level problem.

According to the literature, for optimal solution from sensitivity analysis, the green split of link 1 is 0.2361 and the objective value is 2400.0 (Yang and Yagar, 1995) after the iteration process. While for optimal solution from Log-MILP, the green split of link 1 is 0.2333 and the corresponding final objective value is 2398.8. Obviously, Log-MILP yields a better objective value than that from sensitivity analysis approach, which is reduced by 0.0500% ($= [2398.8 - 2400.0] / 2400.0$) than the result of the former method. Besides, the computational process only needs 0.21 min, which is very fast for solving this non-convex problem.

6.4.1.3 An example of discrete link capacity improvement

It is shown that the results of the CNDSSP with discrete link capacity enhancement on the same test network in Figure 6.3. Suppose each lane added possesses the capacity of 10 units and at most two more lanes can be added to each link. The proposed Log-MILP method is applied to solve the problem and

the partition scheme remains the same as is used in the above tests. The results of optimal discrete capacity improvements are exhibited in Table 6.4. The optimal solution of this case study involves the addition of two more lanes to each of link 1, 2 and 5. The other roadway segments will not be expanded. The green split of link 1 is changed to 0.4043. This optimal solution results in no queuing delay on any signal-controlled link of the network. The value of the objective function achieves 2169.8, which is slightly higher than the result of the CNDSSP model with continuous capacity enhancement. This is because the discrete levels of capacity improvement impose tighter constraints on the structure of the network and thus lead to the objective value not as good as that of the continuous CNDSSP model. However, the adjustment of signal setting can offset part of this effect and make the objective value better. The gap between the objective function value from Log-MILP method and the exact objective value is only 0.0092% ($= [2169.8 - 2169.6] / 2169.6$), which is equal to the gap in the continuous capacity enhancement situation. The computational time of this example is 4.39 min.

Table 6.4 Results of the CNDSSP with discrete capacity enhancement applying the Log-MILP method.

Variable	Value
Green split of link 1	0.4043
Capacity enhancement	
Δs_1	20
Δs_2	20
Δs_5	20
Objective value	
F	2169.8
F_r	2169.6

6.4.2 Case 2: the Nguyen-Dupuis network

Figure 6.4 The Nguyen-Dupuis network.

Table 6.5 Link parameters for the Nguyen-Dupuis network.

Link	Free-flow time	Expansion cost factor	Link	Free-flow time	Expansion cost factor
1	5	3	11	3	2
2	2	2	12	2	2
3	2	3	13	1	4
4	3	3	14	1	4
5	2	2	15	4	1
6	3	4	16	2	3
7	1	6	17	1	7
8	2	2	18	5	1
9	1	4	19	2	5
10	2	6			

To further illustrate the performance of the presented CNDSSP model and the Log-MILP solution method, the second case is applied on a larger-scale network. It is the Nguyen-Dupuis network, as shown in Figure 6.4, which is the same as was used in Nguyen and Dupuis (1984). The network has total four O-

D pairs, 19 links and 25 paths. There are three signal-controlled intersections 6, 9 and 11 that are marked in the network and six signal-controlled links labeled from 1 to 6. The link parameters including free-flow travel time and expansion cost factors are listed in Table 6.5. Table 6.6 shows the demands of all O-D pairs and maximum storage capacity multipliers of links with signal-control. Let all current link capacities be 40 and the maximum capacity of each link after expansions be 60. For the situation with discrete levels of capacity enhancements, it is assumed that each lane possesses ten units of link capacity and maximum two more lanes can be added to each link. Let the minimum green split for each signal-controlled link be 0.1 and the maximum be 0.9. The conversion factor from travel time to money is set to be one and the upper bound of queuing delay of each link with signal is supposing to be ten.

Table 6.6 Demand and maximum storage capacities of signal-controlled links of the Nguyen-Dupuis network.

Maximum storage capacity multiplier: $Q_a(S_a) = k_a \cdot S_a$	Value	Demand	Value
k_1	2.3	1→2	34
k_2	1.7	4→2	46
k_3	1.8	1→3	44
k_4	2.0	4→3	26
k_5	2.2		
k_6	1.9		

In this example, the three variables x_a , s_a and B_a are discretized into 40 equal segments, and λ_a and d_a into twelve equal increments. Table 6.7 illustrates the results solved from the Log-MILP method for the two scenarios with continuous and discrete capacity enhancements. For the case of continuous link

capacity improvements, the direct objective function value and the exact one are 3003.4 and 3101.4 respectively. The gap between them is 3.16% ($= [3003.4-3101.4]/3101.4$). This gap is larger than that in Yang-Yagar example, which may be due to the coarse resolution of variable partition scheme. For the discrete levels of link capacity improvements, the objective function value directly from the Log-MILP is 3061.5 and the exact objective value is 3116.7. The gap (1.77%) is smaller than the continuous scenario. In addition, the computational times for solving the continuous and the discrete case are 10.41 h and 12.30 h respectively.

6.5 Concluding remarks

This chapter proposed an urban transport planning problem (i.e. CNDSSP) with combined network design and signal setting. Spillback phenomenon is considered in saturated conditions by controlling the queuing length on roadway segments to avoid blocking upstream links. A bi-level program model is developed to formulate this problem and then is transformed into an equivalent single-level MIP. In order to obtain a global optimal solution, an efficient mixed-integer linearization technique is applied to cast the problem into a MILP formulation so that the solution possesses the property of global optimization. Finally, numerical experiments from the literature have been carried out and compared with the NDP and the ENSS models and a local optimal solution algorithm. Results show that all the tested cases can be solved efficiently via the commercial optimization software CPLEX. The optimal total system travel cost of CNDSSP formulation is less than those from pure NDP and pure ENSS model. In addition, the proposed global optimal solution

method obtains better urban network design and signal setting plan than a local optimal algorithm.

Table 6.7 Results generated from the Log-MILP method with continues and discrete capacity enhancements.

Variable	Continues capacity enhancement	Discrete capacity enhancement
Green split		
λ_1	0.5874	0.4563
λ_3	0.3207	0.1896
λ_5	0.6311	0.6229
Capacity enhancement		
Δs_1	20	20
Δs_2	20	20
Δs_3	20	20
Δs_4	20	20
Δs_5	19.9834	20
Δs_6		20
Δs_8	20	20
Δs_{10}	15.4667	20
Δs_{13}	9.2310	10
Δs_{15}	20	20
Δs_{18}	20	20
Queuing delay		
d_3	0.3802	3.0119
d_6	4.5345	2.7815
Objective value		
F	3003.4	3061.5
F_r	3101.4	3116.7

Chapter 7 CONCLUSIONS AND FUTURE WORK

This chapter summarizes the research questions and major contributions of this thesis. To advance the models and the global optimal solution methods for network design problem, possible extensions for further research are also discussed.

7.1 Thesis summary

The work studied in this thesis concerns with urban network design problems and their global optimal solution methods. A number of interesting problems in urban transportation network were considered and corresponding mathematical programming models were developed. Since the inherent non-convex feature of NDP, to guarantee the *best* solution, global optimal solution methods were proposed to solve each developed model. A brief summary of the thesis is given as follows.

Chapter 1 introduced the background and the motivation of this thesis. Objectives and outline of the thesis were also stated in this chapter.

Chapter 2 reviewed the literature on the urban transport network design problems and the development of solution algorithms for solving these problems.

Chapter 3 studied a continuous network design problem with stochastic user equilibrium, specifically, logit route choice model, which is a more general and realistic model to describe the road users' route choice behaviour and includes deterministic user equilibrium flow pattern as a particular case. The main contributions of this chapter were two-folded. First, this study filled the

research gap that no global optimization solution method was developed for CNDP with SUE. Furthermore, the solution method developed in the study ensured several good properties of the proposed algorithm, firstly, by applying mixed integer outer approximation approach and a global optimization method, it guaranteed exact global optimal solution property of the original CNDP with SUE, rather than only optimum of the linearized approximation of the original CNDP, and the proof of the global optimality was provided in this chapter; secondly, by using range reduction technique, it ascertained high solution efficiency, which was illustrated in the numerical studies conducted.

Chapter 4 addressed the discrete network design problem with traffic assignment principle of stochastic user equilibrium. The contributions of this chapter were mainly on the algorithm part, since in the literature only an approximated algorithm was provided for the problem. Two global optimization solution algorithms were developed to solve the mixed-integer nonlinear nonconvex model of the DNDP with SUE: one was based on the branch-and-bound method and the other added extra constraints in each iteration to update the iterative process when searching and approaching the global optimal solution. Various linearization and convexification techniques were applied to formulate a convex relaxation of the original model, which provided the lower bound solution of the original problem and served as the key component of two proposed global optimization solution algorithms. Furthermore, instead of using MILP to approximate the nonlinear functions in the original model, which was the spirit of some previous global optimal solution algorithms for continuous network design problem, a convex mixed-integer nonlinear programming was applied to relax the original model. Also an exact global optimal solution of the original model rather than only an approximated global optimal solution could be guaranteed via the proposed global optimal solution algorithms.

Chapter 5 proposed a novel discrete network design problem, which simultaneously considered both the link addition decision (0 or 1) and the optimal link capacity (the optimal link capacity could be either continuous or at discrete levels, both are considered in this chapter) when new links were planned to be added to the existing network. In summary, this chapter contributed to state-of-the-art in the following two aspects. First, this study formulated the optimal new link addition and the optimal road capacity design of the newly added links in an integrated manner, which was not addressed in previous studies in the literature or otherwise considered separately. Second, a global optimization solution algorithm, employing techniques like Reformulation-Linearization techniques, range reduction and outer approximation, was developed to solve the model formulation.

Chapter 6 addressed an urban transport planning problem (i.e. CNDSSP) involving combined network design and signal setting. Spillback phenomenon was considered in saturated conditions by controlling the queuing length on roadway segments to avoid blocking upstream links. This study contributed to the literature in the following aspects. Firstly, a combined network design and signal setting transport planning model was proposed to take into account their mutual interactions and ensure the true optimal transportation planning. Secondly, a global optimal solution method was developed to seek the globally optimal design of the transportation planning problem by applying an efficient mixed-integer model, i.e., Log model, to provide piecewise-linear approximation of the non-linear terms in the reformulated single-level model.

7.2 Recommendation for future work

This thesis answers a number of questions on urban network design problem and how to obtain their globally optimal solution. Nevertheless, as with many studies, the questions raised in this thesis are more than it addressed. Many more questions still need to be answered in this research field. To advance models and global optimal solution methods for network design problems, several issues for follow-up research are presented below.

Firstly, it should be pointed out that, although the property of the global optimality is guaranteed for the proposed solution methods, more efforts should be made to improve the solution efficiency to reduce the computational time for the large-size problem, especially for real transportation network. Since presented algorithms in this thesis can be regarded as frameworks of global optimization solution method, they can be updated by applying more advanced techniques in the future study. For example, chapter 3, chapter 5 and chapter 6 adopted mixed-integer models in the linearization process, which were incorporated in reformulated models thus directly affected the efficiency of solution algorithm. Hence, different mixed-integer models for linearization should be tested in the network design problem for improving the performance of global optimal algorithm. Besides, algorithm in chapter 3 and chapter 5 could also be updated by adopting more efficient range reduction technique.

Secondly, chapter 6 provided a mixed-integer linear approximation for the proposed nonconvex model, the formulation of which relied on the piecewise-linear approximation of the nonlinear terms. To improve the approximation accuracy, except for simply increasing the number of breakpoints, another method is employing a better partition scheme. Hence, advanced partition

method, which better describes the profile characteristics of the graph of nonlinear functions, can be further studied in the future.

Thirdly, the network design problems addressed in this thesis were very general. Specific situations or special cases were not considered in the thesis, for example, elastic demand and link flow constraint. These problems could be further addressed by extending presented model formulations to the more specific situation or the special case.

Last but not least, the methodology and framework used in NDP study could also be extended to investigate similar problems in other transport planning and operation area, such as shipping network design, ship fleet management and operation, ship sailing speed optimization, transit service schedule design, which essentially can be cast into nonlinear and nonconvex programming problems, and the related research works could be presented in future study.

Appendix A: A simple example of outer approximation based linear relaxation

A log function $\ln(y_a)$ is used as a small example to illustrate the application of the outer approximation based linear relaxation. The current feasible range of y_a is $[2, 8]$ and the number of breakpoints is two and thus $n = 4$ (include two endpoints), which is the same as the chart shown in Figure 3.1. Once the number of breakpoints and the bounds of variable is know, the coordinate of these breakpoints can be calculated via Eq. (3.34), that is, $y_a^1 = \underline{y}_a = 2$, $y_a^2 = 4$, $y_a^3 = 6$, $y_a^4 = \bar{y}_a = 8$. The number of binary variable is determined as $\zeta = \lceil \log_2(n-1) \rceil = 2$ via Eq. (3.42). Thus, the linear relaxation is expressed as follows according to Eq. (3.35)-(3.41) and (3.43).

$$l_{y,a} \leq \ln(y_a^1) - 1 + \frac{y_a}{y_a^1},$$

$$l_{y,a} \leq \ln(y_a^2) - 1 + \frac{y_a}{y_a^2},$$

$$l_{y,a} \leq \ln(y_a^3) - 1 + \frac{y_a}{y_a^3},$$

$$l_{y,a} \leq \ln(y_a^4) - 1 + \frac{y_a}{y_a^4}$$

$$l_{y,a} \geq l_{y,a}$$

$$y_a^1 - M \cdot A_1 \leq y_a \leq y_a^2 + M \cdot A_1,$$

$$y_a^2 - M \cdot A_2 \leq y_a \leq y_a^3 + M \cdot A_2,$$

$$y_a^3 - M \cdot A_3 \leq y_a \leq y_a^4 + M \cdot A_3$$

$$\begin{aligned}
& \ln y_a^1 + \frac{\ln y_a^2 - \ln y_a^1}{y_a^2 - y_a^1} \cdot (y_a - y_a^1) - M \cdot A_1 \leq l_{y,a} \\
& \leq \ln y_a^1 + \frac{\ln y_a^2 - \ln y_a^1}{y_a^2 - y_a^1} \cdot (y_a - y_a^1) + M \cdot A_1
\end{aligned}$$

$$\begin{aligned}
& \ln y_a^2 + \frac{\ln y_a^3 - \ln y_a^2}{y_a^3 - y_a^2} \cdot (y_a - y_a^2) - M \cdot A_2 \leq l_{y,a} \\
& \leq \ln y_a^2 + \frac{\ln y_a^3 - \ln y_a^2}{y_a^3 - y_a^2} \cdot (y_a - y_a^2) + M \cdot A_2
\end{aligned}$$

$$\begin{aligned}
& \ln y_a^3 + \frac{\ln y_a^4 - \ln y_a^3}{y_a^4 - y_a^3} \cdot (y_a - y_a^3) - M \cdot A_3 \leq l_{y,a} \\
& \leq \ln y_a^3 + \frac{\ln y_a^4 - \ln y_a^3}{y_a^4 - y_a^3} \cdot (y_a - y_a^3) + M \cdot A_3
\end{aligned}$$

$$A_1 = u_1 + u_2, \quad A_2 = 1 + u_2 - u_1, \quad A_3 = 1 + u_1 - u_2$$

$$j = 1 + u_1 + 2u_2, \quad 1 \leq j \leq 3, \quad u_1, u_2 \in \{0, 1\}.$$

Appendix B: Data for Sioux-Falls network in Chapter 3

Table B.1

Link capacity of Sioux-Falls network

Link	Capacity	Link	Capacity	Link	Capacity	Link	Capacity
1	25.9002	20	7.8418	39	5.0912	58	4.8239
2	23.4034	21	5.0501	40	4.8765	59	5.0026
3	25.9002	22	5.0458	41	5.1275	60	23.4034
4	4.9581	23	10.0000	42	4.9247	61	5.0020
5	23.4034	24	5.0501	43	13.512	62	5.0599
6	17.1105	25	13.9157	44	5.1275	63	5.0756
7	23.4034	26	13.9157	45	15.6508	64	5.0599
8	17.1105	27	10.0000	46	10.3149	65	5.2299
9	17.7827	28	13.5120	47	5.0458	66	4.8853
10	4.9088	29	5.1334	48	5.1334	67	10.3149
11	17.7827	30	4.9935	49	5.2299	68	5.0756
12	4.9479	31	4.9088	50	19.6798	69	5.2299
13	10.0000	32	10.0000	51	4.9935	70	5.0000
14	4.9581	33	4.9088	52	5.2299	71	4.9247
15	4.9479	34	4.8765	53	4.8239	72	5.0000
16	4.8985	35	23.4034	54	23.4034	73	5.0785
17	7.8418	36	4.9088	55	19.6798	74	5.0912
18	23.4034	37	25.9002	56	23.4034	75	4.8853
19	4.8985	38	25.9002	57	15.6508	76	5.0785

Table B.2

Other parameters of Sioux-Falls network for the test in Subsection 3.4.1

Link	(6,8)	(10,16)	(13,24)	Construction cost function	ρ	θ	R_a
$\alpha (\times 10^6)$	2.4864	5.9064	4.0784	$g(y) = \alpha (y - \underline{y})$	1	0.005	0.15
$\underline{y}$	9.7242	7.5213	8.6954				

Table B.3

Upper bound of candidate link capacity in Sioux-Falls network for the test in Subsection 3.4.3

Link	16	17	19	20	25	26	29	39	48	74
$\bar{y}$	10.7242	13.7333	10.7242	13.7333	19.6494	19.6494	12.9721	11.5024	12.9721	11.5024

Appendix C: Data for Sioux-Falls network in Chapter 4

Table C.1

Existing link capacities of Sioux-Falls network.

Link	Capacity	Link	Capacity	Link	Capacity	Link	Capacity
1	25.9002	21	5.0502	40	4.8765	59	5.0026
2	23.4035	22	5.0458	41	5.1275	60	23.4035
3	25.9002	23	10.0000	42	4.9248	61	5.0026
4	4.9582	24	5.0502	43	13.5120	62	5.0599
5	23.4035	25	13.9158	44	5.1275	63	5.0757
6	17.1105	26	13.9158	45	15.6508	64	5.0599
7	23.4035	27	10.0000	46	10.3150	65	5.2299
8	17.1105	28	13.5120	47	5.0458	66	4.8854
9	17.7828	30	4.9935	49	5.2299	67	10.3150
10	4.9088	31	4.9088	50	19.6799	68	5.0757
11	17.7828	32	10.0000	51	4.9935	69	5.2299
12	4.9480	33	4.9088	52	5.2299	70	5.0000
13	10.0000	34	4.8765	53	4.8240	71	4.9248
14	4.9582	35	23.4035	54	23.4035	72	5.0000
15	4.9480	36	4.9088	55	19.6799	73	5.0785
17	7.8418	37	25.9002	56	23.4035	75	4.8854
18	23.4035	38	25.9002	57	15.6508	76	5.0785
20	7.8418	39	5.0913	58	4.8240		

Table C.1

Other input data for Sioux-Falls network.

Candidate link	16	19	29	48	74	Ra
Capacity	4.8986	4.8986	5.1335	5.1335	5.0913	0.15
Cost	325	330	510	500	425	

Appendix D: Updating approximation of VI constraint

Suppose $F = (x_a, t_a, u_a, p_a)$ is the solution to the relaxed VI problem Eq. (5.61), thus F also meets the original VI constraints Eq. (5.42), i.e., for arbitrary link flows $x'_a \in \Psi$, the following inequation (C.1) is satisfied.

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

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

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

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

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

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

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]

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

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 (C.1) is satisfied for all feasible link flows $x_a' \in \Psi$, that is, F is also the solution to the original VI, if
$$\sum_{a \in A} T_a x_a + \sum_{a \in A} T_a R_a p_a - \sum_{a \in A} t_a x_a'^* - \sum_{a \in A_2} (1 - u_a) M \cdot x_a'^* \leq 0 .$$

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 Ψ .

Appendix E: A simple example of branching scheme for equivalent J_1 triangulation

Considering a nonlinear two-dimensional function $z = f(x, y), x \in [0, 8], y \in [1, 3]$ that needs to be linearized applying Log model, the entire domain of (x, y) is partitioned into $4 \times 2 \times 2$ triangles as shown in Figure D.1.

Figure D.1 A triangulation of the domain compatible with J_1

For the first index set $(kx, ky) \in K_1$, when $kx=1$, i.e. kx refers to the x dimension, we have $ky \in \{1, \dots, \lceil \log_2 4 \rceil\} = \{1, 2\}$; when $kx=1$, i.e. kx refers to the y dimension, we have $ky \in \{1, \dots, \lceil \log_2 2 \rceil\} = \{1\}$. This implies that the gray codes for intervals in x direction have two bits, whereas for intervals in y direction the gray codes have only one bit. Assume $G^1 = (0, 0)$, $G^2 = (1, 0)$, $G^3 = (1, 1)$ and $G^4 = (0, 1)$ for $kx=1$; $G^1 = 0$ and $G^2 = 1$ for $kx=2$.

When $kx=1$ and $ky=1$, for points with $x=0$, since their index number $n=0$ and $G_1^1=0$, then all of them belong to $R_{(1,1)}$; for points with $x=2$, since $G_1^1=0$ and $G_1^2=1$, they neither belong to $L_{(1,1)}$ nor $R_{(1,1)}$; for points with $x=4$, since $G_1^2=1$ and $G_1^3=1$, they all belong to $L_{(1,1)}$; for points with $x=6$, since $G_1^3=1$ and $G_1^4=0$, they neither belong to $L_{(1,1)}$ nor $R_{(1,1)}$; for points with $x=8$, since $G_1^4=0$ and $n=N_1=4$, they belong to $R_{(1,1)}$.

Therefore, for index set K_1 , we have:

$$\begin{aligned} L_{(1,1)} &= \{(4,1), (4,2), (4,3)\} \\ R_{(1,1)} &= \{(0,1), (0,2), (0,3), (8,1), (8,2), (8,3)\} \end{aligned}$$

Similarly,

$$\begin{aligned} L_{(1,2)} &= \{(6,1), (6,2), (6,3), (8,1), (8,2), (8,3)\} \\ R_{(1,2)} &= \{(0,1), (0,2), (0,3), (2,1), (2,2), (2,3)\} \\ L_{(2,1)} &= \{(0,3), (2,3), (4,3), (6,3), (8,3)\} \\ R_{(2,1)} &= \{(0,1), (2,1), (4,1), (6,1), (8,1)\} \end{aligned}$$

For the second index set $(kx, ky) \in K_2$, since $kx, ky \in \{1, 2\}$ and $kx < ky$, we have $K_2 = \{(1, 2)\}$. $L_{(1,2)}$ is a set including those points with an even x index number and an odd y index number. $R_{(1,2)}$ is a set including points with an odd x index number and an even y index number.

Therefore, for index set K_2 , we have:

$$\begin{aligned} L_{(1,2)} &= \{(0,2), (4,2), (8,2)\} \\ R_{(1,2)} &= \{(2,1), (2,3), (6,1), (6,3)\} \end{aligned}$$

PUBLICATIONS ARISING FROM THE THESIS

Journal papers:

1. 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.
2. Wang, D.Z.W., Liu, H., Szeto, W.Y., 2015. A novel discrete network design problem formulation and its global optimization solution algorithm. *Transportation Research Part E* 79, 213-230.
3. Liu, H., Wang, D.Z.W., 2015. Modeling and solving discrete network design problem with stochastic user equilibrium. *Journal of Advanced Transportation* (*under review*).
4. Liu, H., Wang, D.Z.W., Yue, H., 2015. Global optimization for transport network expansion and signal setting. *Mathematical Problems in Engineering* 2015.
5. Wang, D.Z.W., Liu, H., Szeto, W.Y., Chow, A.H.F., 2016. Identification of critical combination of vulnerable links in transportation networks – a global optimisation approach. *Transportmetrica A: Transport Science* 12, 4, 346-365.

Conference papers:

1. Liu, H., Wang, D.Z.W., 2013. Global optimization approach for combined traffic assignment and signal control, *Transportation Research Board 92nd Annual Meeting*, Washington, D.C..

2. Liu, H., Wang, D.Z.W., 2014. A global optimization method for linearized urban transport network planning: capacity expansion and signal setting. Extended abstract accepted for presentation in the hEART 2014 conference, Leeds.

REFERENCES

- Abdulaal, M., LeBlanc, L.J., 1979. Continuous equilibrium network design models. *Transportation Research Part B* 13 (1), 19-32.
- Allsop, R.E., 1974. Some possibilities for using traffic control to influence trip distribution and route choice, *Proceedings of the 6th International Symposium on Transportation and Traffic Theory* Elsevier Publishing Company, Incorporated, University of New South Wales, Sydney, Australia, pp. 345-373.
- Bonami, P., Biegler, L.T., Conn, A.R., Cornuéjols, G., Grossmann, I.E., Laird, C.D., Lee, J., Lodi, A., Margot, F., Sawaya, N., Wächter, A., 2008. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization* 5 (2), 186-204.
- Boyce, D., 1984. Urban transportation network-equilibrium and design models: Recent achievements and future prospects. *Environment and Planning A* 16 (11), 1445-1474.
- Breckman, J., 1956. Encoding circuit, United States.
- 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.
- Cascetta, E., Gallo, M., Montella, M., 1998. Optimal signal setting on traffic networks with stochastic equilibrium assignment, *Proceedings of TRISTAN III Symposium*, San Juan, Puerto Rico.
- Ceylan, H., Bell, M.G.H., 2004. Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. *Transportation Research Part B* 38 (4), 329-342.
- Chen, M., Alfa, A.S., 1991. A network design algorithm using a stochastic incremental traffic assignment approach. *Transportation Science* 25 (3), 215-224.
- Chiou, S.-W., 2005. Bilevel programming for the continuous transport network design problem. *Transportation Research Part B* 39 (4), 361-383.
- Cipriani, E., Fusco, G., 2004. Combined signal setting design and traffic assignment problem. *European Journal of Operational Research* 155 (3), 569-583.
- Connors, R.D., Sumalee, A., Watling, D.P., 2007. Sensitivity analysis of the variable demand probit stochastic user equilibrium with multiple user-classes. *Transportation Research Part B* 41 (6), 593-615.
- Croxton, K.L., Gendron, B., Magnanti, T.L., 2003. A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science* 49 (9), 1268-1273.
- Currie, J., Wilson, D.I., 2012. Opti: Lowering the barrier between open source optimizers and the industrial matlab user. *Foundations of Computer-Aided Process Operations*, Savannah, Georgia, USA, 8-11.
- Dafermos, S., 1980. Traffic equilibrium and variational inequalities. *Transportation Science* 14 (1), 42-54.

- Davis, G.A., 1994. Exact local solution of the continuous network design problem via stochastic user equilibrium assignment. *Transportation Research Part B* 28 (1), 61-75.
- Drezner, Z., Wesolowsky, G.O., 2003. Network design: Selection and design of links and facility location. *Transportation Research Part A* 37 (3), 241-256.
- Ekström, J., Rydergren, C., Sumalee, A., 2013. Solving a mixed integer linear program approximation of the toll design problem using constraint generation within a branch-and-cut algorithm. *Transportmetrica A: Transport Science* 10 (9), 791-819.
- Ekström, J., Sumalee, A., Lo, H.K., 2012. Optimizing toll locations and levels using a mixed integer linear approximation approach. *Transportation Research Part B* 46 (7), 834-854.
- 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.
- 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.
- Ferrari, P., 2002. Road network toll pricing and social welfare. *Transportation Research Part B* 36 (5), 471-483.
- Friesz, T.L., 1985. Transportation network equilibrium, design and aggregation: Key developments and research opportunities. *Transportation Research Part A* 19 (5-6), 413-427.
- Friesz, T.L., Anandalingam, G., Mehta, N.J., Nam, K., Shah, S.J., Tobin, R.L., 1993. The multiobjective equilibrium network design problem revisited: A simulated annealing approach. *European Journal of Operational Research* 65 (1), 44-57.
- Friesz, T.L., Cho, H.-j., Mehta, N.J., Tobin, R.L., Anandalingam, G., 1992. A simulated annealing approach to the network design problem with variational inequality constraints. *Transportation Science* 26 (1), 18.
- Friesz, T.L., Tobin, R.L., Cho, H.-J., Mehta, N.J., 1990. Sensitivity analysis based heuristic algorithms for mathematical programs with variational inequality constraints. *Mathematical Programming* 48 (1), 265-284.
- 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.
- Gartner, N.H., Gershwin, S.B., Little, J.D.C., Ross, P., 1980. Pilot study of computer-based urban traffic management. *Transportation Research Part B* 14 (1-2), 203-217.
- Gori, S., Cipriani, E., 2000. Signal control and assignment joint models, *Traffic and transportation studies*, pp. 614-621.
- Heydecker, B.G., Khoo, T.K., 1990. The equilibrium network design problem, *Proceedings of AIRO '90 Conference on Models and Methods for Decision Support*, Sorrento, Italy, pp. 587-602.
- IBM ILOG, 2009. User's manual for cplex, New York.
- Keha, A.B., de Farias Jr, I.R., Nemhauser, G.L., 2004. Models for representing piecewise linear cost functions. *Operations Research Letters* 32 (1), 44-48.

- 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.
- Larsson, T., Patriksson, M., 1995. An augmented lagrangean dual algorithm for link capacity side constrained traffic assignment problems. *Transportation Research Part B* 29 (6), 433-455.
- LeBlanc, L.J., 1975. An algorithm for the discrete network design problem. *Transportation Science* 9 (3), 183-199.
- Lee, C., Machemehl, R.B., 1998. Genetic algorithm, local and iterative searches for combining traffic assignment and signal control, *Proceedings of International Conference on Traffic and Transportation Studies*, Beijing, China, pp. 488-497.
- Lee, C., Machemehl, R.B., 2005. Combined traffic signal control and traffic assignment: Algorithms, implementation and numerical results, University of Texas at Austin, pp. 1-289.
- Lee, J., Wilson, D., 2001. Polyhedral methods for piecewise-linear functions i: The lambda method. *Discrete Applied Mathematics* 108 (3), 269-285.
- 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.
- Li, H.L., Lu, H.C., 2009. Global optimization for generalized geometric programs with mixed free-sign variables. *Operations Research* 57 (3), 701-713.
- 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.
- Lo, H.K., 1999. A novel traffic signal control formulation. *Transportation Research Part A* 33 (6), 433-448.
- Lo, H.K., 2001. A cell-based traffic control formulation: Strategies and benefits of dynamic timing plans. *Transportation Science* 35 (2), 148-164.
- 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.
- Luathap, 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.
- Luo, Z.-Q., Pang, J.-S., Ralph, D., Wu, S.-Q., 1996. Exact penalization and stationarity conditions of mathematical programs with equilibrium constraints. *Mathematical Programming* 75 (1), 19-76.
- Magnanti, T.L., Wong, R.T., 1984. Network design and transportation planning: Models and algorithms. *Transportation Science* 18 (1), 1-55.
- Marcotte, P., 1983. Network optimization with continuous control parameters. *Transportation Science* 17 (2), 181.
- Meneguzzer, C., 1997. Review of models combining traffic assignment and signal control. *Journal of Transportation Engineering* 123 (2), 148-155.
- 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.

- Miandoabchi, E., Farahani, R.Z., Dullaert, W., Szeto, W., 2012. 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.
- Morrison, S.A., 1986. A survey of road pricing. *Transportation Research Part A* 20 (2), 87-97.
- Nguyen, S., Dupuis, C., 1984. An efficient method for computing traffic equilibria in networks with asymmetric transportation costs. *Transportation Science* 18 (2), 185-202.
- 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.
- Sheffi, Y., Powell, W.B., 1983. Optimal signal settings over transportation networks. *Journal of Transportation Engineering* 109 (6), 824.
- Sherali, H.D., 2001. On mixed-integer zero-one representations for separable lower-semicontinuous piecewise-linear functions. *Operations Research Letters* 28 (4), 155-160.
- 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.
- Sumalee, A., Watling, D.P., Nakayama, S., 2006. Reliable network design problem: Case with uncertain demand and total travel time reliability. *Transportation Research Record: Journal of the Transportation Research Board* 1964, 81-90.
- Suwansirikul, C., Friesz, T.L., Tobin, R.L., 1987. Equilibrium decomposed optimization: A heuristic for the continuous equilibrium network design problem. *Transportation Science* 21 (4), 254-263.
- Tan, H.N., Gershwin, S.B., Athans, M., 1979. *Hybrid optimization in urban traffic networks*, Cambridge, MA, p. 123.

- Teklu, F., Sumalee, A., Watling, D., 2007. A genetic algorithm approach for optimizing traffic control signals considering routing. *Computer-Aided Civil and Infrastructure Engineering* 22 (1), 31-43.
- Tobin, R.L., Friesz, T.L., 1988. Sensitivity analysis for equilibrium network flow. *Transportation Science* 22 (4), 242-250.
- Todd, M.J., 1977. Union jack triangulations, in: Karamardian, S., García, C.B. (Eds.), *Fixed points: Algorithms and applications*. Academic Press, New York, pp. 315-336.
- Van Vuren, T., Van Vliet, D., 1992. *Route choice and signal control: The potential for integrated route guidance*. Ashgate Publishing Company, Brookfield, VT, USA.
- Vielma, J., Nemhauser, G., 2011. Modeling disjunctive constraints with a logarithmic number of binary variables and constraints. *Mathematical Programming* 128 (1), 49-72.
- Vielma, J.P., Ahmed, S., Nemhauser, G., 2010. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Operation Research* 58 (2), 303-315.
- Wang, D.Z.W., Liu, H., Szeto, W.Y., 2015. A novel discrete network design problem formulation and its global optimization solution algorithm. *Transportation Research Part E* 79, 213-230.
- 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.
- Webster, F., 1958. Traffic signal settings, p. 44.
- 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.
- Yang, H., Huang, H.J., 2005. *Mathematical and economic theory of road pricing*.
- Yang, H., Yagar, S., 1995. Traffic assignment and signal control in saturated road networks. *Transportation Research Part A* 29 (2), 125-139.
- Ying, J.Q., Lu, H., Shi, J., 2007. An algorithm for local continuous optimization of traffic signals. *European Journal of Operational Research* 181 (3), 1189-1197.
- Zhang, X., van Wee, B., 2012. Enhancing transportation network capacity by congestion pricing with simultaneous toll location and toll level optimization. *Engineering Optimization* 44 (4), 477-488.