

EVACUATION NETWORK OPTIMIZATION: MODELS, SOLUTION METHODS
AND APPLICATIONS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Chi Xie

August 2008

© 2008 Chi Xie

EVACUATION NETWORK OPTIMIZATION: MODELS, SOLUTION METHODS AND APPLICATIONS

Chi Xie, Ph.D.

Cornell University 2008

On finding the most effective ways to minimize the traffic congestion and disaster threat over an urban or regional evacuation network, the focus of this study is to develop a set of analytical tools and computational methods for seeking optimal allocation of existing network capacity and connectivity. The core problem posed in this text is a network optimization problem with regard to two lane-based planning strategies: lane reversal on roadway sections and crossing elimination at intersections. These strategies supplement one another by increasing capacity in specific traffic directions and creating an interruption-free traffic environment throughout the network.

The joint consideration of these strategies greatly increases the problem complexity and combinatorial effect. A Lagrangian-relaxed, tabu-based solution method has been developed to solve this otherwise intractable problem, which takes advantage of Lagrangian relaxation for problem decomposition and complexity reduction and whose algorithmic design is based on the principles of tabu search metaheuristic.

The requirement of emergency vehicle assignment is also incorporated into the above modeling and solution framework, which creates a bi-objective evacuation network optimization problem. A lexicographic optimization approach is developed to identify

the Pareto-optimal set of routing and network solutions for scenario analysis and decision making.

The set of evacuation planning models and solution methods have been tested and evaluated with both numerical examples and an evacuation case study in Monticello, Minnesota with varying network settings and conditions. The evaluation results prove the applicability, reliability and robustness of the developed methodology in both theoretical and practical network circumstances and provide useful insights and directions for further research.

Keywords: Evacuation planning, lane reversal, crossing elimination, network optimization, discrete and combinatorial optimization, multi-objective optimization, Lagrangian relaxation, tabu search

BIOGRAPHICAL SKETCH

Chi Xie obtained his Bachelor of Engineering in Civil Engineering at Tsinghua University, Beijing, China, in the year of 1998. His graduate study and research was focused on Transportation Engineering and he was awarded with a Master of Engineering at National University of Singapore in 2001 and a Master of Science at the University of Massachusetts Amherst in 2005. He has since been pursuing a Doctor of Philosophy in Civil and Environmental Engineering at Cornell University, with two minor areas of Operations Research and Applied Economics. His current research interests include transportation emergency management, transportation system design and planning, transportation network economics and intelligent transportation systems. In addition to his academic activities and experiences, he also provided consultancy services for a number of federal and state government-sponsored transportation planning projects in the U.S. His name has been listed in Marquis Who's Who in America since 2007.

Chi Xie was married to Li Sheng, a communication scientist, on June 27, 2004.

Dedicated to victims of disasters and catastrophic events in history

ACKNOWLEDGMENTS

I owe my deepest gratitude to my advisor, Dr. Mark Turnquist, for his guidance and encouragement throughout my journey of pursuing the doctorate. His knowledge and insights about transportation research as well as broader science areas greatly inspired my interests and shaped my way in thinking of and discovering novel research problems. He is one of the greatest researchers and educators I admire.

I am so grateful to Dr. Linda Nozick, who has always been willing to help me with her expertise and passion when I encountered difficulties in learning and exploring. Her attitude towards work and life brought me a great deal of confidence when I was struggling hardships during the long run of my doctoral study. Her advice has become and shall still be an invaluable asset to my career. My greatest appreciation also goes to Dr. Huseyin Topaloglu, who served on my dissertation committee and introduced me in his course to the joy of a new scientific domain of describing our world—simulation—a way that I had never thought so powerful before. Needless to say, this simulation course is one of the most challenging and rewarding courses I have ever taken during my whole graduate study period.

I would like to extend my gratitude to Dr. Arnim Meyburg, Dr. John Mbwana, Dr. Ningxiong Xu, Dr. June Ma and Dr. Oliver Gao, who shared with me their invaluable experiences and provided me with many pieces of constructive advice at different stages of my doctoral pursuit. I want to acknowledge my fellow colleagues at Cornell, Carmen Rawls, Yashoda Dadkar, Ali Mohammad, Kabehe Vaziri, Pantea Vaziri, Yao Sun, Darrell Sonntag and Brian Levine. I really enjoyed the time of working and

talking with them. One of the most helpful people I met and must mention here is Patty Apgar, who was always accessible and helped me numerous times to catch deadlines in the past years. I am indebted to her.

Special thanks go to Dr. Shashi Shekhar at the University of Minnesota, who shared with me the data sets of the Monticello evacuation network, and Dr. Travis Waller at the University of Texas at Austin, who helped create a visiting researcher position for me so that I could use the university's research facility to continue my dissertation research during the final year of my doctoral study. Without their help, this dissertation could not have been accomplished.

Last but not least, I want to express my deepest appreciation to my wife, Li Sheng, for standing by me through thick and thin, for always believing in me, and for her unfailing love.

TABLE OF CONTENTS

BIOGRAPHICAL SKETCH	iii
DEDICATION	iv
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS	vii
LIST OF FIGURES	x
LIST OF TABLES	xiii
CHAPTER 1 Introduction	1
1.1 The context of evacuation practices	2
1.2 Evacuation travel characteristics	5
1.3 Problem statement	7
1.4 Thesis outline	12
CHAPTER 2 Evacuation planning and optimal network design	15
2.1 Evacuation planning models	15
2.1.1 Simulation-based models	16
2.1.2 Optimization-based models	22
2.1.2.1 Spatial planning	23
2.1.2.2 Temporal planning	26
2.1.2.3 Network planning	27
2.1.3 Concluding remarks	30
2.2 Network design models and solution methods	32
2.2.1 Introduction	33
2.2.2 Model formulations and variations	37
2.2.3 Solution methods	52
2.2.3.1 Exact methods	53
2.2.3.2 Approximate methods	66
2.2.4 Summary	79
CHAPTER 3 Model formulations	81
3.1 Network representation and notation	81
3.2 Travel behavior	85

3.3 System objectives	88
3.4 An evacuation network optimization model	93
3.5 An integrated evacuation network optimization and emergency vehicle assignment model	99
CHAPTER 4 Integrated Lagrangian relaxation and tabu search	104
4.1 Problem complexity	104
4.2 Lagrangian relaxation framework	107
4.3 Tabu search metaheuristic	112
4.3.1 Search neighbors and moves	114
4.3.2 Elite candidate list	120
4.3.3 Tabu list and aspiration criterion	123
4.3.4 Intensification and diversification	125
4.3.5 Lagrangian multiplier updating	128
4.3.6 Stopping rule	129
4.4 Estimation of the network flow pattern	129
4.5 Examination of the intersection crossing elimination	132
4.6 The algorithmic procedure	137
4.7 A numerical example	140
CHAPTER 5 Algorithm calibration and evaluation	147
5.1 Experimental problem instances	148
5.2 Local search calibration	156
5.3 Diversification search calibration	174
5.4 Performance evaluation	174
CHAPTER 6 Evacuation planning for a nuclear power plant	185
6.1 The evacuation network setting	186
6.2 Evacuation network solution development and result analysis	191
6.3 Comparative evaluations	201
6.3.1 Egress availability	201
6.3.2 Demand level	206
6.3.3 Alternative solution methods	210
6.4 Integrated evacuation network optimization and emergency vehicle assignment	217
6.4.1 Emergency vehicle routing	218
6.4.2 Scenario analysis of evacuation network optimization and emergency vehicle assignment	224
CHAPTER 7 Conclusions and further research	233

APPENDIX A Stochastic network loading	239
A.1 The stochastic network loading procedure	239
A.2 Loop deadlock	241
A.3 Covariance of arrival times	244
APPENDIX B Minimizing the number of intersection crossing points	247
B.1 Problem statement	247
B.2 Solution algorithm	249
B.3 Numerical examples	258
BIBLIOGRAPHY	263

LIST OF FIGURES

FIGURE 1.1	Examples of joint lane-reversal and crossing-elimination settings	10
FIGURE 3.1	Intersection subnetwork and roadway-section subnetwork	82
FIGURE 3.2	A performance comparison of minimizing total evacuation time and minimizing the maximum of individual evacuation times	89
FIGURE 3.3	Temporal evacuation generation distributions	91
FIGURE 4.1	Intersection subnetwork reduction	111
FIGURE 4.2	A local move candidate selection procedure	120
FIGURE 4.3	The algorithmic procedure of the LR-TS heuristic	138
FIGURE 4.4	Network information and iterative solutions of the numerical example	141
FIGURE 4.5	Iterations of the optimal traffic movement configuration in the example intersection subnetwork	144
FIGURE 5.1	The problem instances used for the algorithmic parameter calibration	150
FIGURE 5.2	The solution quality and the tabu tenure and elite capacity	164
FIGURE 5.3	The computation efficiency and the elite capacity	166
FIGURE 5.4	Example solution itineraries of the search procedure	169
FIGURE 5.5	Comparison of the objective function values of the optimized solutions and the original configurations	176
FIGURE 5.6	The optimized solutions of the example networks	178
FIGURE 6.1	The emergency planning zone for the Monticello nuclear power plant	186
FIGURE 6.2	The Monticello evacuation network	187

FIGURE 6.3 Traffic crossing points of alternative intersection/interchange designs	191
FIGURE 6.4 Search itineraries for the Monticello evacuation network optimization	193
FIGURE 6.5 Optimized solutions of the Monticello evacuation network	196
FIGURE 6.6 The network performance variation with the egress availability	203
FIGURE 6.7 Variation of the total evacuation time over different demand levels	208
FIGURE 6.8 A network performance comparison between the optimized and sub-optimized network solutions at alternative demand levels	209
FIGURE 6.9 The algorithmic procedures of the SPT and FHFE methods	211
FIGURE 6.10 The Monticello network solutions from the SPT and FHFE methods	214
FIGURE 6.11 A solution quality comparison of three solution methods	217
FIGURE 6.12 Locations of the hospitals inside and around the emergency planning zone	220
FIGURE 6.13 Candidate emergency vehicle routes	221
FIGURE 6.14 The hospital capacity and the emergency route travel time	223
FIGURE 6.15 Optimized evacuation networks and emergency vehicle routes	225
FIGURE 6.16 The Pareto-optimal sets of evacuation network optimization and emergency vehicle routing solutions	231
FIGURE A.1 An illustrative example of the loop deadlock problem	243
FIGURE B.1 The node-arc network representation of a four-leg intersection	247
FIGURE B.2 The transportation tableau representation of the intersection optimization problem	250
FIGURE B.3 Representation of a basic feasible solution in the network and the tableau	251

FIGURE B.4 Feasible region and objective function component of a pair of flow variables with a potential crossing point	256
FIGURE B.5 The first numerical example and its solutions by the simplex-based algorithm	259
FIGURE B.6 The second numerical example and its solutions by the simplex-based algorithm	261

LIST OF TABLES

TABLE 5.1 The calibration results with regard to the elite capacity and tabu tenure	160
TABLE 6.1 List of hospitals and medical centers inside or around the emergency planning zone	222

CHAPTER 1

INTRODUCTION

The journey of a thousand miles starts from where one stands.
—Lao Tzu

Warnings of natural and man-made hazardous events in the U.S. happen at least once a day and there are numerous resulting evacuations each year (Golding and Kaspersen, 1988). Natural disasters include, for example, hurricanes, earthquakes, floods, tornadoes and so on; man-made hazardous incidents may be in principle identified as two types, technological failures and intentional malevolence. Technological fires, hazardous material spills, and nuclear radiation accidents, to name a few, belong to the technological failures; intentional malevolence typically refers to terrorist attacks. Evacuation, as an intuitive and practically effective emergency rescue measure, has long been used and is expected to be enhanced to protect human populations against hazardous situations caused by these natural and man-made disasters.

When the state of technology permits accurate prediction or detection of disastrous events, for example, hurricanes or tornadoes, evacuation is an effective pre-impact tool for reducing the threat from the hazard; when predictions are not feasible, as in the case of fire or terrorist attack, evacuation still serves a variety of emergency functions as a post-impact measure (Lindell and Perry, 1991). A well-defined and manageable plan is one of the prerequisites for successful implementation of a large-scale evacuation. The purpose of an evacuation plan is to maximize the utilization of an existing transportation system for evacuation of a threatened population and hence

to minimize the exposure and potential fatalities of the population from the impending or occurring disaster.

1.1 The context of evacuation practices

The last three decades have seen significant research effort in developing methodologies and implementing technologies for emergency evacuation planning. In the U.S., interest in evacuation planning was invigorated by the nuclear power plant accident occurring at Three Mile Island, Pennsylvania in March 1979. As part of a series of governmental responses, a number of research projects concerning evacuation modeling and clearance time estimation for nuclear radiation emergencies and other hazardous events have been sponsored by government agencies, such as the Federal Emergency Management Agency (FEMA), the Nuclear Regulatory Commission (NRC) and the U.S. Army Corps of Engineers.

More recently, Hurricane Katrina, the third worst hurricane in U.S. history, struck the Gulf Coast in August 2005, resulting more than 1,800 deaths and estimated \$81.2 billion loss of property[†]. The evacuation prior to and just after this hurricane is widely regarded as a failure of emergency response and management. Several post-disaster investigations have cited evacuation failures as a major contribute to the death toll in the city of New Orleans.

[†] Source: The United States Congress (2006). A Failure of Initiative: Final Report of the Select Bipartisan Committee to Investigate the Preparation for and Response to Hurricane Katrina. Government Printing Office, Washington, D.C. Accessed at <<http://www.gpoaccess.gov/katrinareport/fullreport.pdf>>, March 2008.

When Hurricane Rita approached the Texas coastline one month later, residents threatened by this hurricane were keenly aware of the disastrous results caused by Hurricane Katrina. The result was the largest emergency evacuation in the U.S. history and perhaps the largest traffic jam. In the city of Houston and its metropolitan area, though the state and local emergency management officials successfully set up and implemented the evacuation routes based on previous hurricane experiences, the roadway system was still overwhelmed by the enormous and unprecedented numbers of fleeing vehicles. The recorded traffic observations showed that the whole state highway system in the vicinity of the city became gridlocked after the evacuation order was announced, and the heavy traffic snarling the roadways lasted for about 48 hours.

When evacuation plans fail, it reduces people's willingness and confidence in following and cooperating with evacuation orders and plans in the future. A recent behavioral survey conducted by the Harvard School of Public Health on 2,029 adults residing in high-risk hurricane areas in eight states found that about one third of the sampled population would choose not to evacuate during a future hurricane period. Among the sampled population, there are 36 percent of the people believing that "evacuation could be dangerous" and 54 percent believing that "roads are too crowded to leave" (Blendon et al., 2006). This survey result emphasizes that the safety and efficiency of current evacuation practices may not have reached the level people expect. In addition, the evacuation effectiveness can be discounted by people's uncooperative behavior and extra challenges may face emergency management professionals in devising and implementing evacuation plans to deal with future disasters.

The hurricane survey and many other post-disaster investigations on evacuation experiences repeatedly confirmed the fact that the current state of evacuation planning and management was not as prepared for such mass emergency situations as had been previously assumed to warrant a safe and reliable evacuation process. The lessons learned from those historical evacuation events consequently caused evacuation managers, planners and researchers to reconsider and reexamine the effectiveness and efficiency of existing evacuation policies and procedures and called for greater efforts on this developing yet immature research subject.

In the past, evacuation planning and management has been viewed as the responsibility of emergency management and law enforcement agencies. While state and local transportation agencies have been involved in evacuation activities to some degree, their work could be usually characterized as peripheral support (Urbina and Wolshon, 2003; Wolshon, et al., 2005). However, there is increasing awareness that evacuation by nature is a transportation activity. Efforts to improve evacuation planning and capacity building have been renewed by local, state and regional transportation authorities. This includes evacuation demand forecasting and management, evacuation traffic analysis and modeling, evacuation routing and network management, application of intelligent transportation systems for evacuation operation and control, etc.

This viewpoint of modeling and improving an evacuation process as a special transportation system has been recognized since late 1990s, and apparently stimulated after two major hurricanes, Hurricanes Georges in 1998 and Floyd in 1999. The lessons learned from the two mass evacuation instances that carried out statewide and state-crossing traffic movements as well as the experiences gained from other large-

scale emergency management cases indicated that many transportation-related efficiency and safety issues would be frequently encountered yet not satisfactorily addressed at the current state of evacuation practice. The need has been called for a higher level of involvement of transportation professionals in planning and managing an evacuation process.

1.2 Evacuation travel characteristics

While evacuation is inherently a transportation process, it creates unique challenges that are not encountered in conventional transportation planning and management experiences. The sole purpose of an evacuation is to seek safety, rather than any other social or productive activities. Thus an evacuation plan must be enacted with the aim of helping the threatened population to escape from the forthcoming or occurring disastrous event to safe areas or helping them reduce the life-threatening risk to a minimum level. This special travel purpose yields different characteristics of travel demand generation, distribution and behavior in an evacuation network from that in a daily commuting traffic network.

Evacuation time, which may be either total evacuation time or network clearance time, is the primary concern of evacuation managers. Total evacuation time refers to the sum of individual evacuation time over the whole evacuating population in a given emergency area. Network clearance time is a more straightforward time indicator, denoting the time it takes to evacuate the last people since the evacuation onset.

Evacuation is a unique, one-time transportation activity under emergency situations, and evacuees may not have sufficient experience and adequate information to make

proper routing and other travel choices. Their travel decisions and other evacuation-related judgments are highly dependent on their own perceptions of the risk, as well as the information transferred from their social networks and the authority. Due to the unpredictability and suddenness of disasters, their perceived and received information is often inaccurate and incomplete. As a result, travel behavior of evacuees is uncertain and disordered compared to their ordinary travel activities.

The amount of generated evacuating travel from a disaster area may reach a surge level in a short time after an evacuation order is announced to the public or a potential threat is perceived by the population. This exceedingly high rate of travel demand often cannot be accommodated by the existing transportation network capacity that was designed for the daily commuting traffic. In some cases, disasters may have damaged the transportation network or cut off some important corridors in the network. The useable network connectivity and capacity may be considerably limited and needs to be re-evaluated.

Many travel choices, such as destination choices, vary with the nature, location, magnitude and strength of a hazardous event. Destinations often are not well understood by evacuees in advance of the occurrence of a disaster and are also subject to change during the course of an evacuation.

The whole evacuation planning process consists of several interrelated components, including delimitation of the emergency planning zone, estimation of the amount and distribution of evacuating demand, identification of shelter locations or safe zones, configuration, coordination and operation of transportation modes and evacuation routes, and so on. An evacuation plan needs to be developed prior to the full

determination of the geographic scale and magnitude of a forthcoming disastrous event. Therefore, evacuation planning must also incorporate the vulnerability and survivability of the available transportation infrastructure and equipment as well as other supplying resources to the variable and uncertain threats of the disaster. This study is not intended to tackle all these issues arising in evacuation modeling and planning. Instead, it focuses on how to utilize an existing urban or regional ground transportation network, so as to enhance the evacuation performance under an integrated evacuation management framework, in which efficiency is our focal point.

1.3 Problem statement

In this study, we propose and formulate an evacuation network optimization problem from the perspective of traffic network operation and control. It can be regarded as a specific short-term, tactic-level network design problem with a goal of seeking an optimal lane-based network configuration, so as to minimize the total evacuation time or network clearance time for a potentially threatened area.

Given a variety of types of emergency contexts, evacuation planning models may be distinguished in terms of their applicable geographical scales and time spans. An emergency situation caused by a fire, for example, may only need an evacuation covering the residents in its neighborhood. On the other hand, the nuclear power industry uses a circle of 10-mile radius surrounding a nuclear power plant as the emergency planning zone for protecting the people against direct exposure to the radioactive plume in case of nuclear power reactor accidents. The largest scale of evacuation may be caused by hurricanes, which may affect very large regions.

Evacuation planning may deal with either short-term or long-term issues. A short-term evacuation plan needs to be enacted quickly, as an emergency response to an identified or predicted hazardous event. A long-term evacuation plan, on the other hand, is generally proposed for a potential emergency area in which some natural disaster may have frequently occurred in history and is expected to occur again in a foreseeable future.

Our proposed model is applicable to short-term evacuations in an urban or regional context. The model focuses on two lane-based network control measures for enhancing the traffic system performance: roadway lane reversal and intersection crossing elimination. The two lane-based measures alter the capacity and connectivity properties of an evacuation network on its roadway sections and at its intersections, respectively.

Lane reversal is not a new concept. The use of lane reversal results in the co-called traffic “contraflow” or “counterflow” operation. It has been early used as a traffic control solution to accommodate the unbalanced traffic demand between the two driving directions of a congested roadway section. A number of lane reversal studies concerning its design, efficiency, feasibility and safety issues can be seen in, for example, MacDorman (1965), Glickman (1970), Hemphill and Surti (1974), and Caudill and Kuo (1983). An update on the development of lane reversal techniques and applications as well as its current state of planning and engineering practices was recently provided by Wolshon and Lambert (2004). In evacuation cases, the traffic direction of the inbound lanes of some designated roadways may be reversed for the overwhelming outbound traffic with the goal of increasing the outbound capacity. This lane-reallocation strategy has been used extensively in the states along the

Atlantic and Gulf Coasts of the U.S. for hurricane evacuations since the first time it was implemented in Georgia during the period of Hurricane Floyd in September 1999 (Urbina and Wolshon, 2003). In state and regional evacuations, contraflow operation is typically applied to the major arteries (i.e., interstate and state highways).

Crossing elimination at intersections has attracted relatively little attention for evacuation planning and management. Cova and Johnson (2003) suggested using this measure as a lane-based routing strategy for emergency evacuations to reduce traffic control delays (e.g., delays due to traffic signals and stop signs) at intersections. The basic rationale of applying the crossing elimination for evacuation is to convert an intersection with interrupted flow situations to an uninterrupted flow facility by prohibiting some turning movements through blocking lane entries and limiting flow directions. Without the stop-and-go traffic control setting, the intersection capacity for those allowable traffic movements is significantly expanded.

The benefits from implementing the intersection crossing-elimination strategy for evacuation management are threefold. First, it is a desirable control measure to increase the traffic throughput capacity at intersections so as to better serve the exceedingly high traffic demand under emergency conditions. Second, it channels traffic flow along certain routes and improves traffic safety under emergency situations. Third, in the case of a post-disaster evacuation, it may become a critical and necessary remedy measure for intersection traffic control when the traffic signal and communication system fails due to widespread power outages. Such a system failure often occurs in the evacuation cases of no-notice disasters. In the aftermath of the 1985 Mexico City earthquake, for example, most of the traffic signals in the city

network were not functional because of the loss of electric power, damage of traffic sensors, and communication interruption (Ardekani and Hobeika, 1988).

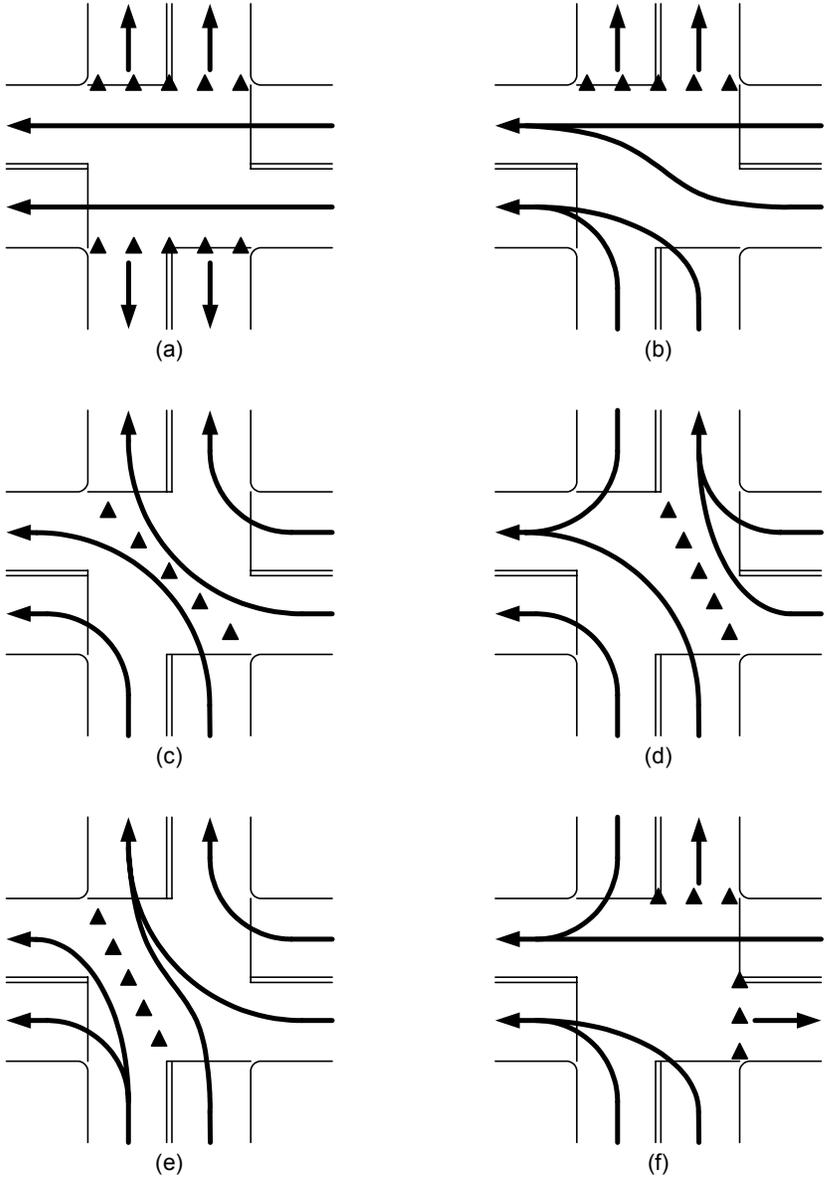


Figure 1.1 Examples of joint lane-reversal and crossing-elimination settings

The two network control strategies may be jointly used to improve the network performance and reduce the evacuation time for urban emergency evacuations. The

integration of these two capacity-increasing strategies has greater potential in optimizing evacuation networks than the application of either of them solely. Several examples of the joint operation of lane reversal and crossing prevention are illustrated in [Figure 1.1](#). Researchers and practitioners have realized the need and importance of such a joint operation in an evacuation network. Tuydes and Ziliaskopoulos (2006), for example, noted in a network contraflow optimization study that the traffic control configurations at intersections and interchanges should be reset to maximize the efficiency of traffic movements resulted from the contraflow operation. However, no study so far has explicitly addressed this integrated evacuation network optimization problem with both the lane-reversing and crossing-elimination settings.

Another important emergency planning requirement is the emergency logistics, or in a simpler manner, the emergency vehicle assignment. Supply, equipment, and emergency management, law enforcement and medical personnel as well as special technical experts need to be promptly transported into the disaster area (Sivanandan et al., 1988). In many cases, although the aerial transportation plays an important role in transporting personnel and resources into the disaster site, its operation is often subject to the insufficient capacity and limited accessibility as well as the clement weather conditions. An efficient emergency logistics system is of the utmost importance to emergency relief efforts under either pre-disaster or post-disaster situations.

However, the current state of practice in the emergency vehicle assignment is far below the satisfactory level. In fact, the most heavily criticized aspect of the official emergency response to hurricanes is the inefficient emergency logistics system (Holgún-Veras et al., 2007). In some hurricane cases, it took up to 2 to 3 weeks to

deliver critical supplies and equipments to the disaster site, which is unacceptable by all the ways.

In view of this requirement, we must reserve one or more inbound routes in the evacuation network to ensure a smooth and efficient ground transportation channel for emergency vehicles. The simplest objective for this requirement may be to find one or more fastest routes and assign the emergency vehicle fleet to these reserved routes. Inevitably, this extra requirement poses a decision-making conflict with the evacuation network optimization objective, in that the emergency vehicle routing demands the roadway capacity with an inbound direction and potentially creates more traffic crossing points with the outbound evacuating traffic at the intersections along the assigned emergency vehicle route.

Given these problem objectives and requirements, our goal is to create tools for effective network evacuation planning. We focus on a detailed network representation where turning movements are represented explicitly (and can be prohibited) and link directions and capacities are treated as decision variables in the model. An optimization model is formulated to represent the problem, and an effective solution method is developed for the optimization. Use of the model is demonstrated in a case study involving evacuation of the area around a nuclear power plant.

1.4 Thesis outline

The remainder of this dissertation is organized as follows. In [Chapter 2](#), relevant previous research work is reviewed. This includes traffic flow and travel choice modeling mechanisms for evacuation networks as well as discrete network design

models and solution strategies, which form the basis for the modeling of optimal network problems presented in this work.

In [Chapter 3](#), we discuss the modeling rationales and problem formulations. The first model is proposed for an evacuation network optimization problem with lane reversal and crossing elimination operations. The second is an extension of the first problem with an additional consideration of emergency vehicle assignment in an evacuation plan. Within these models, we elaborate the network representation, travel behavior assumptions, system objectives, and model structure for the evacuation network optimization and emergency vehicle assignment problems.

Solving these optimal evacuation network problems poses a very challenging computational task. [Chapter 4](#) presents an integrated Lagrangian relaxation and tabu search heuristic to solve these otherwise intractable problems. This method takes advantage of Lagrangian relaxation for problem decomposition and complexity reduction and its algorithmic design is guided by the principles of the tabu search metaheuristic. An illustrative problem is provided to interpret the rationale and effectiveness of the proposed solution method.

The algorithmic procedure of the proposed solution method is then calibrated and tested in [Chapter 5](#) with several synthetic and realistic evacuation networks. In [Chapter 6](#), we then apply the model to solve a realistic evacuation problem for the area surrounding a nuclear power plant. The effectiveness and efficiency of the solution procedure is evaluated and compared to selected methods in the literature.

Finally, modeling and computing experiences from this study are summarized in [Chapter 7](#). Some concluding remarks and suggestions for future research are also included.

CHAPTER 2

EVACUATION PLANNING AND OPTIMAL NETWORK DESIGN

All experience is an arch to build upon.
—Henry Brooks Adams

The literature review presented in this chapter can be grouped into two focus areas. The first part summarizes emergency evacuation planning methods and applications, discusses the problems encountered in developing and applying the existing models, and finally evaluates the gap between the latest research progress and the demanding requirements from the present evacuation problems. The second part synthesizes research results from previous network design studies, providing modeling and algorithmic insights as well as computational experiences for developing the problem formulation and solution strategies in this study.

This joint discussion of different research areas provides the technical platform of the modeling and solution methodologies used in this research.

2.1 Evacuation planning models

Regional evacuation planning models may be categorized into two types: optimized-based and simulation-based models. An optimized-based model is typically of the functional form of a network flow or design problem and can be directly used to search for the optimal evacuation plans. This approach tells “what to” do in making an evacuation plan. Simulation-based models function as a “what if” methodology

and can be used to evaluate a set of pre-specified candidate plans. In the following subsections, we discuss simulation approaches first, followed by optimization models.

2.1.1 Simulation-based models

Evacuation planning must be based on the behavioral intentions of the individuals who perceive themselves at risk (Johnson and Zeigler, 1983). Emergency evacuation is a complex, interactive travel process between the overwhelmingly large evacuating demand and the relatively scarce transportation supply. The behavioral characteristics of individual evacuees, e.g., their responses to the emergency situations and their behaviors in travel choices, may be difficult to be precisely quantified in an analytical way. Therefore, simulation, which is capable of modeling an evacuation process in a detailed, disaggregated and distributed manner, has been used in evacuation planning tool. In what follows, a number of specific evacuation simulation models are briefly described in a chronological order, and the evacuating behavior modeling mechanisms within these models are highlighted.

The earliest emergency evacuation simulator arising in the literature may be NETVAC (Network Emergency Evacuation), due to Sheffi et al. (1981, 1982). NETVAC is a macroscopic evacuation traffic simulation model, which was originally designed for modeling traffic flow patterns and estimating clearance times in hazardous events caused by nuclear power plant accidents. The evacuation-specific traffic modeling mechanisms in NETVAC include its queue formation and route selection processes, both of which are modeled at an aggregate level. Some commonly used evacuation management strategies, for example, intersection controls and lane operation measures, were included as the evacuation planning options in customizing planning scenarios.

To take into account the abnormal routing behaviors in emergency situations, the authors specified the route choice probabilities with the arriving drivers at an intersection in terms of a combination of two factors, a prior knowledge of the network directionality and a “myopic” observation on the traffic conditions on the outbound roadways directly ahead. Such a behavioral assumption implies that drivers choose their evacuating routes based on how fast a series of outbound links can get them out of the emergency area. The authors also considered the capacity calculation and intersection control issues with some adjustments to reflect the highly congested and chaotic evacuation conditions.

IDYNEV (Interactive DYNamic EVacuation) is a multi-module evacuation model with the functions of assessing evacuation plans and estimating evacuation travel times, as developed by KLD Associates for the Federal Emergency Management Agency (FEMA). It consists of three functional modules: traffic assignment module, traffic simulation module and traffic capacity module. The capacity module serves both the assignment and simulation modules with the function of estimating the roadway capacities, considering turning movements, geometrics and other factors. The assignment module functions in estimating traffic routes in terms of the static user-equilibrium principle. The simulation module is used to mimic the dynamic traffic movements based on the assigned routes from the assignment module as well as make necessary rerouting if the assigned routes are overly congested. The basic simulation mechanism of this module is actually an adaptation of the TRAFLO[†] simulation model with some extensions in scope to accommodate all types of facilities.

[†] TRAFLO (later called CORFLO) is a macroscopic traffic simulation package developed by FHWA in 1970s.

An application of IDYNEV for developing evacuation plans for five nuclear power stations is described in FEMA (1984).

As a core evacuation modeling module, IDYNEV has also been included in several larger evacuation planning and emergency mitigation systems sponsored by FEMA, such as EESF (Exercise Evaluation and Simulation Facility) (FEMA, 1984), IEMIS (Integrated Emergency Management Information System) (Meitzler et al., 1986; Bower et al., 1990), and PCDYNEV (PC-based DYnamic Network EVacuation) (Goldblatt and Weinisch, 2005).

Hobeika and his colleagues (Hobeika and Jamei, 1985; Hobeika et al., 1994) devised a macroscopic simulation-based evacuation planning model named MASSVAC (MASS eVACuation), in which flow status and propagation is described by the analytical traffic flow relationships. This model later became the core module in an evacuation planning software package—TEDSS (Transportation Evacuation Decision Support System) (Hobeika et al., 1994). By incorporating previous investigations on people’s responses to an emergency warning or evacuation order, MASSVAC suggests a logistic *S*-shape curve to model the cumulative demand loading into an evacuation network. In its latest version, there are two user-specified options for modeling traffic assignment: stochastic logit-based flow assignment method and deterministic user-optimal flow assignment method. Moreover, to better reflect the evacuees’ routing behavior, all paths containing any link leading toward the disaster source are eliminated in implementing either of the assignment methods. Such a mechanism is set by a simple “distance” model and “angle” model (see Hobeika and Kim, 1998 for details).

Stern and Sinuany-Stern (1989) and Sinuany-Stern and Stern (1993) described a microscopic simulation model, SNEM (SLAM[†] Network Evacuation Model), for an effort of elaborating evacuating behaviors. Their concerns on the behavioral aspect not only include the evacuees' routing behaviors, but also their responses to the emergency warning and their preparation activities. There are three main components in their simulation model, where the first one is to generate household activities and the next two are to simulate traffic flows at roadway sections and intersections respectively. In the household module, they developed a tree diagram to simulate residents' evacuation response and decision behavior so that the diffusion time of evacuation instructions and preparation time for evacuation can be estimated from the simulation result. As for route selection, the authors simply assumed that an evacuee would choose a shortest path from his location to a closest egress of the disaster area. If the evacuee observes at any intersection the next link on his pre-specified shortest path is full of vehicular queue, he will choose the next shortest path for the remaining trip.

An alternative macroscopic evacuation simulation model similar to NETVAC was developed by Han (1990) for accommodating the use of public transportation in evacuation. This simulation model resides in an evacuation decision support system called TEVACS (Transportation EVACuation System). The basic simulation logic and mechanism in TEVACS are actually the same as that in NETVAC. For example, both of them use two major logical components, a link process and a node process, to model the traffic flow characteristics and evacuees' travel choices. However, TEVACS has its unique features in simulating public transit and mixed traffic flow, in

[†] SLAM stands for Simulation Language for Alternative Modeling. It is a simulation language that provides a unified system modeling framework, which allows systems to be simulated from the perspective of a process, event, or state variable.

which the deployment of public evacuation routes and gathering points can be explicitly specified and the use of different transportation modes such as buses, automobiles, motorcycles and bicycles can all be considered in an evacuation process.

OREMS (Oak Ridge Evacuation Modeling System) is a simulation-based software package that allows for comprehensive evacuation planning studies, such as development of traffic management and control strategies, identification of evacuation routes, estimation of evacuation times, and others (Rathi and Solanki, 1993). The core component is a macroscopic traffic flow simulation model named ESIM (Evacuation SIMulations). The traffic flow modeling principles implied in ESIM are based on the platoon dispersion theory and its implementation was directly adapted from NETFLO II[†]. ESIM uses a combined destination and route choice procedure for the evacuation flow assignment. The destination choice behavior is modeled in a hybrid form of three types: 1) evacuees will follow a given evacuation plan to select a designated evacuation exit; 2) evacuees will choose a closest exit in terms of his pre-perceived static roadway conditions; and 3) evacuees will choose a closest exit in terms of the concurrent dynamic roadway conditions. The combination of destination and route choice is realized by hypothesizing a hypothesized destination and all the exits of the disaster area connecting to this destination via dummy links. The evacuating trip distribution and traffic assignment is carried on by the user-equilibrium assignment method, destined to the hypothesized destination. With each of such dummy links, an impedance traveling cost is assumed, which is used to adjust the relative attractiveness of the evacuation exists in the combined trip distribution and traffic assignment process.

[†] NETFLO II is a macroscopic traffic simulation model included in the TRAFLO package, which was developed by FHWA in 1970s.

Pidd et al. (1996), Pidd et al. (1997) and de Silva and Eglese (2000) presented a prototype of a microscopic simulation model known as CEMPS (Configurable Emergency Management and Planning System), which is a spatial decision support system built on a geographic information system (GIS) platform. In this integrated decision support system, the simulation model is used to model the evacuation process while the GIS platform is used to manage and manipulate the geographical and infrastructural data and visualize the simulation results. As similar to other evacuation simulation models, the current version of CEMPS also assumes that route search is a myopic process in that drivers would choose their ways to the destinations by taking account of immediate congestion conditions ahead.

Several other efforts have applied either one of the evacuation-specific simulators or general traffic simulation models to specific evacuation problems. Some examples are (e.g., Radwan et al. (1985)[#], Southworth and Chin (1987)^{*}, Cova and Johnson (2002)^{**}, Church and Sexton (2002)^{**}, Chen and Zhan (2008)^{**}, Jha et al. (2004)[§], Theodoulou

[#] Radwan et al. (1985) used NETSIM (NETwork SIMulation) to simulate the emergency evacuations in a rural network in Blacksberg, Virginia. NETSIM is a microscopic traffic simulator developed by the Federal Highway Administration (FHWA), which was specifically designed for simulating surface street networks.

^{*} Southworth and Chin (1987) applied MASSVAC (MASS eVACuation) to simulate an evacuation case caused by flooding as a result of dam failure. MASSVAC is a macroscopic evacuation simulation model developed at the Virginia Polytechnic Institute and State University. It will be introduced later in this text.

^{**} Cova and Johnson (2002) and Church and Sexton (2002) both simulated a neighborhood-scale evacuation using PARAMICS (PARAllel MICROscopic Simulator) in the context of wildfire emergency evacuation. Chen and Zhan (2008) applied PARAMICS to evaluate the effectiveness of staged evacuation strategies in a set of hypothesized and realistic traffic networks. PARAMICS is a microscopic traffic simulator developed by Quadstone Ltd., U.K.

[§] Jha et al. (2004) used MITSIM (MICROscopic Traffic SIMulator), the core simulator in a traffic network analysis software suite called MITSIMLab, to simulate an evacuation network located in White Rock and Las Alamos, New Mexico. MITSIMLab is a traffic simulation system developed at Massachusetts Institute of Technology, which was designed to evaluate traffic management strategies at the operational level.

and Wolshon (2004)[†], Murray-Tuite and Mahmassani (2004)[‡], Kwon and Pitt (2005)[‡] and Yuan et al. (2006)[‡]). These case studies are more focused on the scenario analysis rather than the model development, which therefore are of our minor interest and will not be discussed here.

It should be realized that the simulation-based evacuation planning models are merely suitable for evaluating and assessing, but not for generating evacuation plans directly. These plans, however, need to be devised by some other external procedure or conceived in terms of the emergency planner's experience and judgment. The optimization-based evacuation planning models, as described below, provide this functionality and capability.

2.1.2 Optimization-based models

Use of traffic simulators to evaluate and select evacuation planning scenarios is somehow a trial-and-error approach. As an alternative paradigm, optimization-based evacuation planning models have the capability of identifying optimal scenarios in a systematic, self-driven manner. The optimization-oriented feature leads evacuation

[†] Theodoulou and Wolshon (2004) took use of CORSIM (CORridor SIMulation) to evaluate and compare several alternative contraflow scenarios on Interstate 10 highway at the outskirts of New Orleans. It consists of two main traffic simulation components, NETSIM (for arterial streets) and FRESIM (for freeways). CORSIM is one of the most widely used microscopic traffic simulator, whose development and maintenance was sponsored by FHWA since 1970s, and now has been in a commercial package called TSIS (Traffic Software Integrated System).

[‡] Murray-Tuite and Mahmassani (2004), Kwon and Pitt (2005) and Yuan et al. (2006) all used DYNASMART-P, the planning version of DYNASMART (DYnamic Network Assignment Simulation Model for Advanced Road Telematics), as the evacuation evaluation tool in their respective studies. Murray-Tuite and Mahmassani modeled and assessed the impact of household-based preparedness trip chains on the evacuation efficiency for the south-central portion of Fort Worth, Texas; Kwon and Pitt tested alternative contraflow and ramp access strategies for the downtown area of Minneapolis, Minnesota; and, Yuan et al. compared a set of scenarios with different route and destination choice settings for Knox County, Tennessee. DYNASMART is a dynamic traffic assignment-simulation model developed for transportation planning and operations analysis under the FHWA's Dynamic Traffic Assignment (DTA) research project.

planning models of this type to be written in a mathematical programming form, whose objective is typically defined as minimization of the total evacuation time, minimization of the network clearance time, or maximization of the network traffic throughput. To cope with the optimization program, a set of constraints reflecting the inherent supply-demand relationships are used to regulate the system behavior.

In terms of evacuation planning components (i.e., decision variables in an optimization-based evacuation planning model), existing optimization-based models may be categorized into three types. The first type of models focuses on the spatial planning of evacuation demand, including the destination selection and route assignment; the second type is concerned with the temporal planning of evacuation demand, such as the demand departure scheduling; the third type is related to the network planning, which, in the context of evacuation planning, can be conducted only on the short-term, tactic level, such as the contraflow operation and intersection control. While the first two types of models are intended to optimize the spatial and temporal distributions of evacuation demand, the third type aims to optimize the supply side of an evacuation system, which in general results in the so-called network design or network redesign models.

2.1.2.1 Spatial planning

In developing an evacuation decision support system, Tufekci and Kisko (1991) proposed a minimax optimization model with the linear structure. This linear programming problem has the flow conservation constraints for each origin-destination (O-D) pair and the inequality constraints indicating that the travel time of each link must be accommodated by the evacuation clearance time. The objective of

this minimax problem is to minimize the network clearance time, which is defined as the maximum of all the link clearance times. The decision variables in the model, path flows, are subject to a user-equilibrium traffic assignment on the K -shortest routes for each O-D pair. The authors assumed that in a given evacuation planning zone evacuees would choose K -shortest routes at most to escape, where the value of K is estimated by the evacuation planners. Therefore, a K -shortest path problem needs to be solved as a subproblem of the clearance time minimization problem.

The authors further considered a dynamic evacuation flow optimization problem in a time-expanded network with a time-accumulative evacuation demand. To save the computational cost in searching for the optimal solution, a heuristic method that is to generate the evacuation flow pattern for each O-D pair separately was developed. In their work, however, no explicit model formulation and solution algorithm were presented.

Sherali et al. (1991) considered an evacuation planning problem with jointly optimizing traffic flow distribution and shelter construction. The shelter locations need to be chosen from a set of given candidate sites. It has obviously the formulation of a typical discrete network location or network design problem. In their model, a central authority is assumed to have the power of controlling the evacuation flow. Meanwhile, congestion effect and capacity setting on the link level are incorporated into the model formulation. To solve this nonlinear mixed integer programming problem, the authors devised two algorithms, one exact algorithm and one approximate algorithm, based on the generalized Benders decomposition technique.

Dunn and Newton (1992) suggested an evacuation planning model with a minimum cost flow problem formulation, where the “cost” is the total evacuation time. Given a static evacuation network, the purpose of their model is to find the optimal flow pattern that minimizes the evacuation time, subject to the pre-specified flow upper limit of each link. They applied the classic out-of-kilter algorithm to solve this minimization problem. For the case of shelters with a capacity, the authors suggested to employ a network transformation strategy, as similar to Rathi and Solanki (1993) described above: a super dummy node and a dummy link connecting each of the capacitated shelters to the super node are hypothesized; the capacities of these dummy links are set equal to the capacities of the corresponding emanating shelter nodes.

Yamada (1996) considered an evacuation planning problem for urban pedestrian flow. In his study, no congestion effect was assumed with the pedestrian flow and the evacuees’ movement speed was estimated as a constant. Given the incapacitated and capacitated types of refuges, he presented two evacuation planning models. In the incapacitated case, the evacuation planning problem collapses to a shortest path problem for each residential area to its closest refuge, which was solved by the well-known Dijkstra’s algorithm; in the capacitated case, the evacuation planning problem turns to be a standard minimum cost flow problem, which was solved by the out-of-kilter algorithm.

A two-level evacuation planning problem was formulated and solved by Liu et al. (2006a) for determining the optimal routing and destination scheme in an evacuation plan. In this optimization model, the upper- and lower-level objectives are intended to maximize the total traffic throughput during a given evacuation period and to minimize the total travel time and waiting time (at origins) if the given duration is

sufficient for evacuating all evacuees, respectively. The underlying traffic flow pattern is specified by a dynamic traffic assignment model based on a revised version of the cell transmission concept and modeling mechanism (see Ziliaskopoulos, 2000; Li et al., 1999).

2.1.2.2 Temporal planning

Spatial planning models are capable of determining the destination and route choices of evacuees, while temporal planning models can further optimize the temporal distribution of evacuation demand by specifying the departure time choice, which result in the so-called staged evacuation planning. With incorporating the time dimension into an evacuation plan, these models are inevitably established on time-dependent networks.

Sbayti and Mahmassani (2006) proposed an evacuation planning model to search for the optimal combination of departure time, route and destination for an evacuation network so as to minimize the network clearance time. They employed a mesoscopic traffic simulator, DYNASMART-P, to mimic the underlying traffic assignment and determine the network state, and developed an iterative heuristic procedure to approximate the optimal temporal distribution of departure times. The sequential staging policy is finally extracted from the continuous departure-time distribution.

In another staged evacuation planning model, Liu et al. (2006b) focused on optimizing the starting time of each evacuation demand zone with a given demand generation rate and temporal pattern in advance. The objective of this model is to minimize the sum of total urgency-weighted evacuation time and waiting time. As similar to their

previous evacuation planning study (Liu et al., 2006a), a dynamic traffic assignment model based on the cell transmission technique was employed to evaluate the system state and the objective function.

The decision variables for describing an evacuation scheduling policy could also include the number of evacuation stages. Chien and Korikanthimath (2007) proposed an analytical method to model the evacuation staging process and estimate the evacuation time and delay for an evacuation corridor with a uniform demand-generation distribution along the length. This method was then used to determine the optimal number of stages with an equal duration length. While there is lack of solution algorithm that can directly determine the optimal staging solution, their study justifies that an improved scheduling interval and zoning range scenario is important in reducing the evacuation time.

2.1.2.3 Network planning

In recent years, devising an optimal evacuation plan has been advanced from seeking the optimal routing and scheduling scheme in a given network and spreading evacuees to follow the corresponding system-optimal evacuation order, to promoting the optimal flow pattern by physically manipulating the network configuration. In these network planning models, the decision variables not only reside in demand routing and scheduling, but also include network supply properties, such as contraflow configuration and intersection control.

Cova and Johnson (2003), by arguing that most traffic delays occur at intersections during an evacuation, introduced the lane-based routing strategy in evacuation

planning, which is of the purpose of limiting the traffic turning movements at intersections. Given the lane-based routing requirement in their proposed model, a feasible evacuation plan must have no crossing conflict between any two directions of traffic flow throughout an intersection. In practice, such intersection control measures can be readily implemented with some temporal installations of traffic barriers and road signs as well as emergency management personnel at intersections. The authors formulated a linear mixed integer programming model, where the constant travel impedance of a link is represented by its distance. This mixed integer programming problem was then solved by a branch-and-bound algorithm. Since distance cannot fully represent the travel time in an evacuation process, the authors also sought and evaluated other solutions that possess a trade-off between reducing the evacuating distance and decreasing the number of flow merges occurring at intersections.

Hamza-Lup et al. (2004, 2007) proposed an evacuation network and route planning model for seeking full contraflow configurations on all eligible links in an evacuation network and planning evacuation routes based on the contraflow configuration. Two simple network optimization heuristics were proposed. The first heuristic determines the lane-reversing direction of each link based on the coordinate information of the two end intersections of the link, that is, an outbound traffic direction is chosen for lane reversal, while the second one determines the lane reversal configuration and evacuation routes by searching for shortest paths from the emergency source node to all exit nodes. The signal control and coordination strategy is also incorporated to enhance the robustness and efficiency of an evacuation routing plan. In their modeling setting, however, the signal optimization process is performed as a subsequent step to the route planning. A possible incompatible issue between the route planning and the signal control may result in the system suboptimality.

Another lane-based evacuation network optimization model was recently developed by Tuydes and Ziliaskopoulos (2006). This study exclusively dealt with contraflow as the planning component in an evacuation network. The contraflow optimization model has an attractive linear programming (LP) formulation, in that a system-optimal objective is assumed with the network evacuating behavior and a dynamic traffic assignment model based on the cell transmission concept is used to describe the evacuating flow pattern. The authors noted the importance of intersection control in compliance with the contraflow assignment, but did not include it in their model. This LP network optimization model can be solved exactly. For an evacuation network of realistic size, however, the problem is very large and the authors resorted to a tabu-based heuristic to search for the optimal contraflow configuration and the VISTA (Visual Interactive System for Transport Algorithms) simulation package (see Ziliaskopoulos and Waller, 2000) was adopted to estimate the system-optimal traffic flow pattern.

Meng et al. (2008) followed a similar approach to define and solve a lane-based contraflow optimization problem. The decision variables in their model are the number of lanes for each traffic direction of the candidate links. The traffic simulator adopted to serve the lower-level problem is PARAMICS (PARAllel MICROscopic Simulator). A genetic algorithm was developed to search for the optimal contraflow solution and the simulator's application programming interface (API) provides the functionality of implementing the contraflow configuration given by the upper-level decision and delivering the feedback information from the lower-level simulation results.

Shekhar and Kim (2006) also considered a contraflow optimization model in the context of discrete network design problems, where their specific setting regarding the contraflow operation is the full lane reversal only and the objective of their model is to minimize the network clearance time. They modeled a dynamic network evacuation process in the framework of capacitated, fixed-cost networks and employed an expanded space-time network to accommodate the time-dependent flow propagation. Their model is in nature a linear mixed integer programming model. To maintain a good trade-off between the solution optimality and computational efficiency, the authors suggested different solution algorithms to address the proposed contraflow optimization problem with different demand levels, where the demand level is evaluated by a term named overload degree that is defined as the ratio of the number of evacuees over the capacity of the network bottleneck. In their paper, exact integer programming algorithms were suggested for solving the network contraflow problem with a low overload degree; a simple greedy heuristic of selecting contraflow links based on the link congestion level in the original network was developed for the problem with an intermediate overload degree; as for the case with an extremely high overload degree, it was regarded as a network optimization problem in a single-source, single-sink network with infinity demand and a heuristic based on the renowned max-flow, min-cut theorem was accordingly proposed to identify the network bottleneck links and to reverse their coupled links for contraflow.

2.1.3 Concluding remarks

To maintain the solution tractability, these evacuation planning models formulated as network flow or network design problems are typically built on the basis of somewhat unrealistic behavioral assumptions, which inevitably result in the lose of precision in

modeling the behavior details of an evacuation process. Despite the ease of implementation and less computational resource required, the main weaknesses of these optimization-based evacuation planning models revealed by this literature review are: 1) a central emergency management authority is implicitly assumed to regulate the whole evacuation process while the individual behaviors may not be appropriately taken into account; and 2) in many cases, a proper interpretation of the congestion effect or the relationship between traffic flow and travel time is not incorporated into the estimation of the spatial distribution of evacuating flow over the network.

It is expected that a joint use of optimization-based and simulation-based models could combine the merits of both sides and enhance the quality of devised evacuation plans. This may be achieved in such a way that optimization-based models generate a set of candidate evacuation schemes and evacuation simulators are then used to conduct a comparative evaluation and make an ultimate recommendation. Such an approach actually has been implied in some of the studies reviewed above, such as Tufekci and Kisko (1991), Sherali et al. (1991), Cova and Johnson (2003) and Liu et al. (2006a). Case studies recently conducted by Zou et al. (2005) and Liu et al. (2008) explicitly used a network optimization module to generate optimal traffic routing, scheduling, lane reversal and signal control strategies and employed the CORSIM and VISSIM simulators to evaluate the generated evacuation plans.

In some cases, the necessity of using simulation for the evaluation of an evacuation system is not only because simulation is capable of mimicking the evacuation process in greater details and accommodating time-varying, stochastic environments, but also because it provides a more comprehensive and reliable performance measure for the

system evaluation. Typical performance measures used in an evacuation optimization model include, for example, the total evacuation time, the network clearance time, and the network traffic throughout. However, none of these performance measures could be a universal criterion in all evacuation planning problems; each of them only focuses on and reflects one of the aspects of the evacuation efficiency and is only applicable to some specific evacuation situation. The evacuation performance could be better assessed by evaluating the number of evacuees having left the disaster area over time. Such an evaluation process requires a simulation model, or in other words, in these cases, an integrated optimization-simulation approach is highly desirable.

A tighter integration of optimization-based and simulation-based models could be realized by inserting a simulation module into a bi-level optimization framework, in which the simulation module constitutes the lower-level problem and is used to evaluate the objective function of the upper-level problem. We have seen such examples in Tuydes and Ziliaskopoulos (2006), Sbayti and Mahmassani (2006) and Meng et al. (2008). Though this optimization-based, simulation-embedded method combines the advantages of the two modeling paradigms, it typically requires extensive computation, especially if the simulation is on the microscopic level.

2.2 Network design models and solution methods

We have defined the optimal evacuation network problem as a lane-based network design problem. The decision variables for the evacuation network configuration include the assignment of lanes on each reversible roadway section, which reflects the lane reversal setting, and the availability and connectivity of intersection turning movements, which indicate the crossing elimination setting. In our evacuation

network optimization model, both of the decision variables are in the discrete form. Therefore, a literature review regarding the development of discrete network design models and their solution methods is presented below.

2.2.1 Introduction

In the transportation research field, the earliest network design models were created by Garrison and Marble (1958) and Quandt (1960). The network design models in both studies were a result of the efforts in extending the standard transportation problem[†] to a linear programming form of transportation network supply problems. The structure of their models can be described as “combining a transportation model with a road construction model”. Quandt (1960) presented several formulations of the network design problem from the simple standard transportation model to the transshipment model, where the latter became the initial prototype of the present problem formulations we widely refer to today.

Network design problems are often called optimal network problems or network optimization problems in the literature, as the objective of a generic network design problem is to seek an optimal cost-effective network topology and capacity solution with taking into account the network infrastructure investment and the resulting network operation efficiency.

[†] By the *standard transportation problem* here, we mean the Hitchcock-Koopmans transportation problem, in which an optimal transportation flow pattern needs to be determined with carrying a single homogeneous good from a group of origins with the known supply capacities to a group of destinations with the known demand amounts.

There have been several criteria to classify network design problems. In terms of the nature of their design decision variables, network design problems can be categorized into the following two groups: 1) discrete network design problems, which deals with adding (and deleting) links to an existing network; 2) continuous network design problems, which, typically, seek the optimal capacity expansion or assignment for a network. This natural discrepancy is so distinct that the discrete and continuous network design problems have been tackled by completely different solution strategies. Of course, mixed network design problems may also be defined, in which both adding new links and improving existing links are considered jointly. The mix of discrete and continuous variables does not alter the essential discrete nature of a mixed network design problem, however, since the level of the continuous variables can be determined in the evaluation of each combination of the discrete variables (Stairs, 1968).

From the perspective of the transportation planning and engineering, a discrete network design model is of more convincing value and better applicability than a continuous model (Steenbrink, 1974a, b), since, for some practical reasons, a physical transportation network capacity construction or expansion is confined to some discrete choices, for example, addition of a new roadway link or deletion of an existing link, and expansion or reduction of the capacity of a link by a certain number of lanes. In accordance to these reasons, a network design problem is often formulated as an integer or mixed integer programming model (Wong, 1985b).

There have been a few state-of-the-art surveys about the modeling and algorithmic development of discrete network design problems. Magnanti and Wong (1984) synthesized many discrete network design models and algorithms as well as their

applications and computational experiences. The majority of the network design models reviewed in their work do not consider the congestion effect. With this assumption, a network design model is set so that individual users (i.e., units of flow) in the network with fixed arc travel costs would have a consistent minimum-cost objective with the system planner or operator. In other words, the system-optimal objective function of such a network design problem implicitly reflects individuals' user-optimal behavior; no extra constraint addressing individual routing behavior needs to be specified. In such a way, these fixed-cost network design problems were typically formulated as linear mixed integer programming models, where the integer decision variables represent discrete choices of the network design components and the continuous variables are network flows.

By incorporating the congestion effect, the model formulation of a network design problem would be changed in two terms: first, the network operation cost function is nonlinear no matter in what form an arc cost function is; second, given an arbitrary network solution, the system-optimal objective is in general not consistent with the user-optimal routing behavior. In a retrospective survey, Minoux (1989) reviewed a set of models and solution strategies for discrete network design problems with different types of congestion effect that arose from a variety of application contexts. His focus was given to variable-cost network design problems, capacitated spanning tree problems and concentrated location problems. The models of using different arc cost functions, such as linear cost, concave cost, piecewise cost, and some mixed cost functions, were discussed in terms of their formulations, solution methods and applications. Yang and Bell (1998), in another review report, presented a comprehensive review on network design models and algorithms with the specific variable-cost setting in the transportation context, including both discrete and

continuous forms. These models are typically formulated as bi-level programming problems, in which the objective functions of the upper-level and lower-level subproblems are respectively to minimize the system cost and to minimize the individual travel cost in response to the upper-level design decision.

Such a bi-level network programming problem can be described by a Stackelberg game in the context of game theory (see Fisk, 1984), in which the game leader is the network planner who takes charge of the network connectivity and capacity assignment and the follower is the population of network users who individually minimize their own travel costs in response to the leader's decision and other competing users' actions.

In this review work, our main concern is discrete network design problems. By tracking representative studies, we intend to present an overall picture of the development of discrete network design models and solution strategies in a unifying framework and explore the possibilities of extending these techniques for current and forthcoming discrete network design applications. To reflect the latest achievement in this field, our focus is given to the bi-level network design problems, which, as we mentioned, are typically referred to the equilibrium network design problems. In what follows, we first present the fundamental problem formulations synthesized from a variety of previous discrete network design studies. Although these models share many common features, they can be distinguished by a number of dimensions, such as the type and magnitude of decision variables, treatment of the network construction budget, consideration of the congestion effect, assumption about the individual routing behavior, and so on.

2.2.2 Model formulations and variations

A discrete network design problem was initially proposed to provide a decision-making tool for the infrastructure investment planning of, typically, production, transportation, distribution, and communication networks. It emerges in nature as a multi-objective optimization problem: to minimize the network operation cost and the capital investment cost (as well as other fixed or variable cost components) simultaneously. For a transportation network, the capital cost may include not only the infrastructure design and construction costs, but also the loss of amenity and damage to the environment. A transportation network design problem may also be extended to include other objectives, for example, maximization of the traffic demand (MacKinnon and Hodgson, 1970)[†], minimization of the total travel distance and minimization of the relocation of residence units (see, for example, Friesz et al., 1993; Friesz, 1981).

With partial efforts that are aimed to relaxing the multi-objective optimization complexity as well as the actual economic pursuit, two typical formulations have been often used. The first formulation assumes an upper bound on the capital cost so that the network design problem becomes a single-objective problem with the aim at minimizing the network operation cost and with a capital cost constraint. If the upper bound is uncertain, several tentative values might need to be used. With each tentative bound value, a specific optimal operation and capital cost combination solution can be derived. The collection of all these solution combinations is then evaluated for the final decision making. The second formulation resorts, when the operation and capital

[†] By arguing that one of the fundamental purposes of a transportation system is to facilitate movements, MacKinnon and Hodgson (1970) formulated an integrated network optimization problem that combines trip generation, distribution and assignment.

costs are commensurable, to summing both the operation and capital cost components into the objective function via a weighting combination. The determination of the weighting coefficients may be subject to an economist's judgment.

For the sake of comparison, we repeat the two typical discrete network design problem formulations here. Suppose that we are given a directed network $G = (N, A)$, N is the set of nodes, and A is the set of arcs connecting the nodes in N . The arc set A consists of two exclusive subsets, a subset of fixed arcs A_F and a subset of variable arcs A_V , where $A = A_F \cup A_V$ and $A_F \cap A_V = \emptyset$. Thus a possible network solution is (N, A_C) , where arc set A_C that satisfies $A_F \subseteq A_C \subseteq A$, that is, arc set A_C at least includes all the arcs in A_F and at most includes all the arcs in A_F and A_V . The objective of a discrete network design problem is to find an optimal arc set A_C that achieves a preset investment criterion. Each element a in A_V is represented by a binary decision variable, z_a , in the discrete network design problem. When $z_a = 1$, it indicates that arc a is included in subset A_C ; when $z_a = 0$, it means that arc a is excluded. Given these graphical and notational settings, the first model formulation can be given as,

$$\min \quad c(\mathbf{x}, \mathbf{z}) = \sum_{a \in A_C} x_a c_a(x_a) \quad (2.1)$$

$$\text{s.t.} \quad \sum_{a \in A_V} d_a z_a \leq B \quad (2.2)$$

$$z_a = 0 \text{ or } 1 \quad \forall a \in A_V \quad (2.3)$$

$$\sum_k f_k^{rs} = b_{rs} \quad \forall r, s \quad (2.4)$$

$$x_a = \sum_{rs} \sum_k f_k^{rs} \delta_{ak}^{rs} \quad \forall a \in A_C \quad (2.5)$$

$$x_a \geq 0 \quad \forall a \in A_C \quad (2.6)$$

$$x_a \leq M_a z_a \quad \forall a \in A_C \quad (2.7)$$

$$\mathbf{x} \in F \quad (2.8)$$

where this model actually present two sets of decision variables, (\mathbf{x}, \mathbf{z}) , which respectively represent the supply and demand performances in the given network $G = (N, A)$: z_a is the 0-1 integer design variable that indicates the choice of a candidate link a , $a \in A_V$, and x_a denotes the resulting flow amount on link a , $a \in A_C$. The objective function of this model denotes the sum of all the network operation costs, where $c_a(\cdot)$ is the arc travel cost function of arc a .

The most commonly imposed constraints for this optimization model include the following. The design budget constraint (i.e., constraint (2.2)) specifies the upper limit of the total investment, B , where d_a is the cost for designing and constructing arc a . The flow conservation constraints (i.e., constraints (2.4)-(2.5)) reserve the sum of path flows, f_k^{rs} , between each origin-destination (O-D) pair $r-s$, where b_{rs} represents the total travel demand between O-D pair $r-s$ and δ_{ak}^{rs} indicates the incidence relationship between arc a and path k . The capacity constraint (i.e., constraint (2.6)) may have different implications for capacitated and incapacitated problems. When the network design problem is incapacitated, this constraint merely indicates the arc availability; when the problem is capacitated, in addition to the arc availability, it also sets up the upper bound of arc flow, where M_a denotes the capacity of arc a . This constraint is often called forcing constraint, in that it imposes the relationship between design variables and flow variables. Finally, we express the user routing behavior constraint (i.e., constraint (2.8)) in an implicit form since different implications may be included by this constraint, where F represents the feasible space of traffic flow

patterns specified by the individual routing behavior, as subject to the network topology and capacity. In most network design models, the user-optimal routing behavior is assumed. However, the interactions between the user routing behavior and other problem settings would lead to different traffic assignment principles, which may be expressed in explicit or implicit form. We will observe these differences in the different versions of discrete network design problems.

The second discrete network design model considers the capital cost in a different way, in which the investment budget constraint is relaxed, and instead, the sum of the network operation cost and capital cost is minimized as the objective function. Its formulation is as follows,

$$\min \quad c(\mathbf{x}, \mathbf{z}) = \sum_{a \in A_C} x_a c_a(x_a) + w \sum_{a \in A_V} d_a z_a \quad (2.9)$$

$$\text{s.t.} \quad z_a = 0 \text{ or } 1 \quad \forall a \in A_V \quad (2.10)$$

$$\sum_k f_k^{rs} = b_{rs} \quad \forall r, s \quad (2.11)$$

$$x_a = \sum_{rs} \sum_k f_k^{rs} \delta_{ak}^{rs} \quad \forall a \in A_C \quad (2.12)$$

$$x_a \geq 0 \quad \forall a \in A_C \quad (2.13)$$

$$x_a \leq M_a z_a \quad \forall a \in A_C \quad (2.14)$$

$$\mathbf{x} \in F \quad (2.15)$$

It can be seen that this second model formulation has exactly the same set of variables and coefficients except that the second model has a coefficient for the capital cost, w , which represents the conversion factor between the network operation cost and capital cost, while the first model has an upper bound on the capital investment, B .

Apparently, the difference between the two model formulations are merely the treatment of the capital cost component.

In fact, these two network design problem formulations show the two facets of a single problem, which can be converted between each other by duality theory or Lagrangian relaxation technique. This conversion can be readily seen by relaxing the capital cost constraint in the second formulation and compensating this relaxation in the objective function with a penalty component, $w(\sum_{a \in A'} d_a z_a - B)$, where w is the well-known dual price or Lagrangian multiplier. If w is properly set to equal the weighting coefficient that converts the capital cost into the operation cost, this relaxed Lagrangian program from the second problem formulation is equivalent to the first formulation.

The two problem formulations, on the other hand, have their respective economic implications in the transportation investment and planning practice. The first model arises when a certain amount of network improvement budget has been approved by the legislature and the transportation planning authority wants to achieve the greatest travel time saving (i.e., the network operation cost) under this budget constraint. The second model depicts a picture that the capital cost will be ultimately paid by the network users through taxes, for which the capital cost in some sense should be part of the minimization.

The two models do not exclude each other, however. In fact, a more general problem formulation could be developed to accommodate both the capital cost treatments in a single model, such as the one shown in Magnanti and Wong (1984). The two formulations can then be regarded as the special cases of this general model. When

the investment budget bound is relatively low, the budget constraint will be tightened at the optimal solution and the operation cost will dominate the objective function. Then the general model will collapse to the first problem formulation. When the budget bound is relatively high, the budget constraint, apparently, will actually recess in constraining the optimization problem, which results in the second problem formulation.

Within either of the model formulations, an optimal solution to the network design problem can be regarded as a trade-off between the network capital and operation costs. The degree of this trade-off is dependent on the value of the budget bound (in the first formulation) or the conversion factor (in the second formulation). In many situations, thus, a sensitivity analysis that involves multiple settings for the budget constraint or the conversion factor may need to be conducted to find a best trade-off between the two parts of the system cost. However, an extreme end of this trade-off is, if a network planner is only concerned about the network operation cost given that the budget limit is sufficient, the network design problem can be relatively readily solved. In its simplest case, which we will discuss later, if we assume that the travel cost on each arc is fixed without the congestion consideration, the optimal network solution is simply the one with all the candidate links to be built. On the other hand, another extreme situation is, if a network planner wants to minimize the capital cost without considering the operation cost, a minimum spanning tree solution should satisfy this optimality criterion.

In addition to the two classic network design models presented above, other formulations could be devised. Hershendorfer (1965), for example, discussed the possibility of a discrete network design model with the objective of minimizing the

capital cost given a minimum travel cost saving requirement; Chan et al. (1989) materialized the formulation of such a model with a system-optimal, congestion-dependent traffic assignment. Although this alternative model has its different economic interpretation, it does not cause any additional theoretical difficulty in the problem formulation and solution development (as compared to the first two models), and its relationship to the second formulation can be similarly analyzed by using the duality property or Lagrangian relaxation.

Moreover, these basic network design models can be further extended to contain some topological or geometric restrictions or flexibilities on candidate arcs, the design components may not be limited to arcs, and the time dimension could be introduced to create a multi-period rather than once-through design problem. Extra restrictions may include, for example, the precedence relationship and the multiple choice relationship (see Magnanti and Wong, 1984). In some urban planning cases, both one-way and two-way roadway sections may be permitted in designing a transportation network (see, for example, Drezner and Wesolowsky, 1997, 2003; Cantarella et al., 2006). A lane-based network design problem is also introduced, in which the lane-based capacity on an eligible roadway section can be distributed between the two traffic directions (see, for example, Meng et al. (2008)). The introduction of the time dimension into a multi-period network design problem creates the dynamic network design problem, such as the one described by Rothengatter (1979). If node selection is included in a discrete network design problem, it typically results in a joint network design and facility location interface, in which facility sites are generally located at nodes and the facility locations are related to the node selection.

The variants of discrete network design models are not only distinguished by the treatment of the network investment budget, but also discerned in terms of other modeling assumptions and settings. Most of the early discrete network design models developed in 1960s and 1970s were originally designed to construct an entirely new network, while many others later on were aimed to seek an optimal expansion scheme to an existing network. While there exists no intrinsic difference between the full and partial network design problems in the methodology regard, a solution procedure for the former problem must include a network feasibility or connectivity test. On the other hand, the later problem may be of greater interest for practical applications in which transportation planning projects nowadays are often to add or expand roadways to an existing network, rather than to construct a new network.

When the capacity is far beyond the expected traffic flow rate in a transportation network, ignoring either the arc capacity or the congestion effect is acceptable. In this case, the traffic routing as well as the calculation of the network operation cost is based on the fixed arc travel cost. However, it is more reasonable in most application cases to recognize the capacity effect.

Depending on different problem contexts and modeling assumptions, the capacity setting could be treated in two ways: capacity-constraint and capacity-restraint. A network design problem of the former type is also called a capacitated problem. In a capacitated network design problem, the capacity of an arc is imposed as an upper limit on the traffic flow rate of the arc (see constraints (2.7) and (2.14)). Many communication, logistics and air transportation systems are often modeled as capacitated networks. The problem formulation and solution strategies of capacitated

problems have been discussed by, for example, Balakrishnan (1984), Lamar (1990) and Gendron et al. (1998).

On the other hand, the capacity-restraint approach sets the capacity in a network design problem in an indirect way, which is more common in ground transportation networks. In a capacity-restraint problem, the capacity effect is achieved by considering the congestion phenomenon and incorporating the congestion-dependent variable travel cost into the network design. As different from a capacity-constraint problem, the arc capacity in a capacity-restraint problem appears as a parameter in its travel cost functions, instead of the upper bounds of the capacity constraints. In accordance with the use of variable travel costs, the (deterministic or stochastic) user-equilibrium routing principle has been often assumed to regulate the traffic flow pattern. Needless to say, the fixed arc cost in either an incapacitated or capacitated case implies that the network operation cost is linear to the traffic flow rate, while the variable arc cost results in the network operation cost in a nonlinear form. It is, of course, possible to consider both the capacity-constraint and capacity-restraint settings in a network design problem with, for example, a user-equilibrium behavior assumption, in which the capacity of an arc plays a dual role that specifies the skewness of the travel cost curve and sets an upper limit on this curve simultaneously. It has been suggested that this setting could better model the arc performance in a transportation network. In the meantime, however, it increases the solution complexity greatly, in that the underlying traffic assignment itself (i.e., a capacitated user-equilibrium assignment problem) is sufficiently complex to solve.

The evaluation of the objective function involves an estimation of the network flow pattern, which is the result of a traffic assignment process. Depending on the different

capacity settings, the traffic assignment routine within a network design problem could be an all-or-nothing assignment, multi-commodity assignment[†], system-optimal assignment, user-equilibrium assignment, and so on. The underlying traffic assignment process is such an important component in a network design model that it not only specifies the structural property and complexity of the model, but also determines the computational efficiency, since a dominant part of the computational cost for a network optimization process is spent in repeatedly evaluating the objective function.

As we have seen, the traffic assignment principle underlying a network design problem is confined by several factors, such as the arc travel cost function, individual routing behavior assumption, and capacity setting. Apparently, a capacitated network involves a higher degree of complexity in the model structure and solution method for both the traffic assignment problem and the network design problem. Here we concentrate on discussing several alterations of the problem formulation caused by the arc travel cost function and routing behavior assumption.

If the arc travel costs are given as pre-fixed values rather than flow-dependent variables, every individual user can choose his route independently without interacting with any other users. Under this assumption, the user-optimal routing behavior results in, on the individual level, only the shortest path to be chosen based on the fixed arc cost and, on the network level, that the traffic flow pattern can be simply specified by the all-or-nothing assignment. Moreover, the user-optimal routing principle is consistent with the system-optimal objective of the network design problem, or in

[†] By the *multi-commodity assignment*, we mean the traffic assignment problem in a capacitated, fixed-cost network. See [Chapter 17](#) in Ahuja et al. (1993) for a formal treatment on this topic.

other words, the system-optimal objective simultaneously takes into account the user-optimal behavior. In accordance, there is no need to impose any form of the routing behavior constraint in the problem formulation (i.e., constraint (2.8) in the first problem formulation or (2.15) in the second formulation can be relaxed). A number of researchers investigated discrete network design problems with this simplest behavior assumption, including, but not limited to, Scott (1967, 1969), Stairs (1968), Ridley (1968), MacKinnon and Hodgson (1970), Boyce et al. (1973), Hoang (1973), Billheimer and Gray (1973), Steenbrink (1974a, b), Pearman (1974), Dionne and Florian (1979), Boffey and Hinxman (1979), Los and Lardinois (1982), and Magnanti and Wong (1984).

In the transportation planning field, the congestion effect is often measured by a capacity-restraint cost function that has a convex functional form (e.g., the widely used Bureau of Public Roads function). By incorporating the congestion effect this way, a discrete network design problem can be described by a nonlinear integer programming model. Under this condition, the user-optimal routing behavior is not consistent with the system-optimal objective. In the literature, one approach to deal with this contradiction is to ignore the user-optimal routing behavior and instead assume all network users to follow a system-optimal routing behavior (e.g., Sherali et al. (1991)). In this case, the problem formulation can still be simplified in that the individual routing behavior constraint is relaxed.

Another approach is to respect the user-optimal routing behavior and hence to write it into the model in an explicit form. However, there exists no convenient closed form of user-optimal behavior constraints. One way to express the user-optimal routing principle is to append a mathematical program. Such a modeling setting results in the

so-called bi-level optimization problem, in which the upper-level problem has the original network design objective and the lower-level problem as a constraint of the upper-level problem is used to describe the user-optimal traffic assignment. LeBlanc (1975) and Hoang (1982)[†] may be the first two who rigorously investigated a discrete network design problem with the congestion effect and with an explicit incorporation of the user-equilibrium routing constraint. In their studies, the lower-level problem (i.e., the user-equilibrium constraint) is specified by Beckmann's mathematical programming form:

$$\min \sum_a \int_0^{x_a} c_a(\omega) d\omega \quad (2.16)$$

s.t. constraints (2.3)-(2.7) (or (2.10)-(2.14))

As we mentioned above, in a given network with the congestion effect, the user-optimal and system-optimal flow solutions in general will not coincide at the same point. A fundamental difficulty thus arises when solving a network design problem with minimizing the system cost while satisfying the user-optimal behavior constraint. It comes as what is commonly called Braess' paradox, which can be described as, for example, an arc addition, which decreases the objective function value of the user-equilibrium problem, may increase the objective function value of the system-optimal problem subject to the user-equilibrium constraint.

[†] Ochoa-Rosso (1968) described a discrete network design problem with the congestion effect and the user-optimal behavior constraint and suggested a branch-and-bound method to solve this problem. His work, however, is not reviewed here because 1) no explicit user-optimal routing formulation was given in his text, 2) the non-convexity property caused by Braess' paradox identified later was not considered, and 3) neither detailed algorithmic procedure nor computation effort was described.

Poorzahedy and Turnquist (1982) proposed an approximate model to the discrete network design problem with the user-equilibrium constraint, by observing that in the case of using a polynomial-type arc cost function, the system-optimal objective function is approximately linear to the user-equilibrium objective function. Their approximation suggested using Beckmann's user-equilibrium objective function (i.e., function (2.16)) to replace the system-optimal objective function (i.e., function (2.1) or (2.9)), which favorably eliminates the bi-level modeling and solution complexity.

To consider the uncertainty in individuals' routing behaviors leads to stochastic traffic assignment methods. Chen and Alfa (1991) proposed a discrete network design problem with a stochastic user-equilibrium behavior constraint. In their model, the logit-based stochastic user-equilibrium assignment model by Fisk (1980) was chosen to specify the lower-level problem:

$$\begin{aligned} \min \quad & \frac{1}{\theta} \sum_{rs} \sum_k f_k^{rs} \ln(f_k^{rs}) + \sum_a \int_0^{x_a} c_a(\omega) d\omega & (2.17) \\ \text{s.t.} \quad & \text{constraints (2.3)-(2.7) (or (2.10)-(2.14))} \end{aligned}$$

As similar to LeBlanc's model, the network design problem with a stochastic user-equilibrium constraint is also susceptible to Braess' paradox, which results in the non-convexity of the search space. The computational complexity as well as the algorithmic challenge caused by these alternative modeling settings will be discussed in the next section.

Another important modeling dimension related to transportation network design problems is to include trip generation and distribution in such a demand-supply

interaction fashion. We have so far only discussed the network design models with a fixed O-D demand pattern. A transportation network change in either topology or capacity, which is the result of a network design and planning process, has long-term, inevitable impacts on travel demand generations and distributions. It is, therefore, desirable to incorporate both the destination and route choice behaviors in a transportation network design problem. Boyce and Soberanes (1979), Boyce and Janson (1980), and Janson and Husaini (1987) discussed a discrete network design problem with a combined trip distribution and traffic assignment. To avoid the contradiction between the objectives of the network design problem and the trip distribution problem, where the latter is in general posed as an entropy maximization problem, they employed a doubly-constrained trip distribution problem formulation proposed by Erlander (1977), which aims to minimize the total travel cost subject to an entropy constraint. All these modeling extensions to this network design problem with the variable trip distribution setting can be summarized as the following constraints appended to the original problem formulation (e.g., problem formulation 1),

$$\sum_s b_{rs} = O_r \quad \forall r \quad (2.18)$$

$$\sum_r b_{rs} = D_s \quad \forall s \quad (2.19)$$

$$-\sum_{rs} p_{rs} \ln(p_{rs}) \geq E_0 \quad \text{where } p_{rs} = b_{rs} / \sum_{rs} b_{rs} \quad (2.20)$$

where O_r and D_s represent the trip production and attraction demands at origin node r and destination node s , respectively, and E_0 is a pre-specified entropy constant. Constraints (2.18) and (2.19) specify the demand conservations at origin and destination nodes; constraint (2.20) represents the entropy requirement. It should be

noted that in these constraints b_{rs} is not a fixed demand number, but a variable indicating the trip number between origin node r and destination node s .

An integrated demand-supply transportation network design model can also be specified by an elastic-demand equilibrium function on its lower level (Yang and Bell, 1998). In this problem formulation, trip generation and assignment are both the results of a network expansion or modification. By assuming the demand between an O-D pair is continuous and decreasing function of the congestion level between this O-D pair, the lower-level problem can be written as (Sheffi, 1985),

$$\min \sum_a \int_0^{x_a} c_a(\omega) d\omega - \sum_{rs} \int_0^{b_{rs}} q_{rs}^{-1}(\omega) d\omega \quad (2.21)$$

$$\text{s.t. } b_{rs} \geq 0 \quad (2.22)$$

constraints (2.3)-(2.7) (or (2.10)-(2.14))

where $q_{rs}^{-1}(\cdot)$ is the inverse of the demand function for O-D pair $r-s$, b_{rs} is a non-negative demand variable rather than a fixed demand rate for O-D pair $r-s$, and other notations are the same as before.

As part of a network optimization case study, MacKinnon and Hodgson (1970) presented an alternative transportation network design model that integrates trip generation, distribution and assignment. In their setting, the trip generation, distribution and assignment processes are all subject to the network connectivity through a gravity model. Thus, an alternative objective function was developed to evaluate the network performance, that is, to maximize the sum of the traffic flow rates over the network:

$$\max \sum_{rs} O_r D_s / c_{rs} \quad \text{where } c_{rs} = \min_k \sum_a t_a \delta_{ak}^{rs}, \forall r, s \quad (2.23)$$

s.t. constraints (2.3)-(2.7) (or (2.10)-(2.14))

where c_{rs} is the travel time of the shortest path between origin r and destination s , and O_r and D_s in this model have a slightly different meaning from the aforementioned definition: they are not the trip production and attraction demands, but represent synthetic socio-demographic factors that potentially produce and attract travel demands, respectively. The actual travel production and attraction demand rates for any O-D pair are functions of a network design solution. Given the fixed-cost assumption, it is known that the traffic flow maximization objective implies that the traffic flow for any O-D pair is always assigned on its shortest path.

2.2.3 Solution methods

Despite the various formulations and different degrees of intricacy, these discrete network design problems pose the NP-hard computational complexity, even in its simplest form. Johnson et al. (1978) establishes its NP-completeness by showing the classic knapsack problem is reducible to the discrete network design problem, where the former is a well-known problem with NP-completeness. In fact, Wong (1980) showed that even finding an approximate discrete network solution is an NP-hard problem.

There have been a number of solution methods, including exact and approximate methods, developed to solve discrete network design problems. Due to the structural

similarity of different versions of discrete network design problems, these methods, each of which was developed and applied for solving a particular set of problems, may, with some minor modifications, be transferred to other discrete network design problems. Thus, in the following, we do not distinguish these methodologies by their specific network design models and applications. Instead, we would rather present the solution algorithms in a synthetic form where their common procedures and features are concentrated. Different algorithmic designs for particular problem cases and modeling settings are then supplemented in a comparative manner.

2.2.3.1 Exact methods

Due to the combinatorial complexity, there are only a limited number of algorithmic choices in devising a mathematical programming method to solve a discrete network design problem optimally and efficiently. The branch-and-bound and Bender's decomposition strategies may be the most effective optimization-based techniques in this field. Both of the techniques are aimed to reducing the number of combinatorial possibilities by setting lower bounds or Benders cuts so that the optimal solution can be exhausted by an enumeration search in a reasonable time frame.

2.2.3.1.1 Branch and bound

Branch and bound is a generic algorithmic paradigm to solve discrete and combinatorial optimization problems. In fact, it is an implicit enumeration strategy that searches all the discrete feasible regions for the optimal solution except those pruned non-optimal parts. The performance of a branch-and-bound procedure is determined by two tools. The first one is a recursive process of partitioning the whole

search space into feasible subspaces by fixing variables to their discrete feasible choices. This whole partitioning process can be naturally represented by a tree structure that consists of vertices and branches. A partitioning step occurs at a vertex[†], dividing the search space associated with the current vertex into a number of exclusive subspaces, each of which corresponds to one discrete value of the selected variable. Such a partitioning process at a vertex is called *branching*. To accelerate the enumeration search, a *bounding* tool is used to eliminate those branch spaces that prove to exclude the optimal solution through comparing the upper and lower bounds to the optimal solution. As we will see below, the branch-and-bound framework offers such a universal approach that it can handle both single-level and bi-level network design problems and it is so flexible as to accommodate a variety of branching and bounding procedures.

One implementation of the branch-and-bound strategy was devised by Boyce et al. (1973) for the discrete network design problem with incapacitated arcs and no congestion effect, which can be stated as follows. Given a branch-and-bound search tree, any vertex represents a search space spanned by three types of arcs: included arcs, excluded arcs, and undetermined arcs, where we use S_I , S_E and S_U to represent the set of included arcs, set of excluded arcs and set of undetermined arcs, respectively. Any network solution belonging to the search space must include all the links in S_I and not include all the links in S_E , and may include part of the links in S_U . According to this definition, we know that at the root vertex, $S_I = A_F$, $S_E = \emptyset$, and $S_U = A_V$. Starting from the root vertex, the branching process progressively expands to construct a search tree. At each branching step, we select an undetermined arc a from S_U and

[†] The term “vertex” used here is referred to a state point of the branch-and-bound tree structure. In contrast, the term “node” is exclusively used in this text to represent an intersection, interchange or terminal of a network.

assign a into S_I and S_E , respectively. This results in two succeeding vertexes, where at one vertex, its corresponding included, excluded and undetermined arc sets become:

$$S_I^1 = S_I \cup \{a\}, S_E^1 = S_E, \text{ and } S_U^1 = S_U - \{a\}, \quad (2.24)$$

and at another vertex, its corresponding arc sets are:

$$S_I^2 = S_I, S_E^2 = S_E \cup \{a\}, \text{ and } S_U^2 = S_U - \{a\}. \quad (2.25)$$

Prior to branching at each vertex, a bounding examination is carried out by comparing the current upper and lower bounds to the optimal solution. The upper bound is the objective function value of the best feasible solution obtained so far, while the lower bound provides a possible lowest objective function value to the whole solution set in the branch space associated with the current vertex. If the upper bound is lower than or equal to the lower bound, the vertex is regarded as a terminal vertex and the succeeding search in the current branch space can be discarded. In such a way, no further branching from this vertex is needed and the enumeration search effort is accordingly lessened. Otherwise, the possibility of a better feasible solution in the current search space still exists and this vertex should be branched into two succeeding vertexes. Particularly, a vertex with $S_U = \emptyset$ is apparently a terminal vertex since it contains only one solution and has no vertex successor. Reaching every terminal vertex in a search tree is equivalent to a full enumeration of the search space. Therefore, the performance of a branch-and-bound procedure is dependent on its algorithmic efficiency that eliminates non-optimal branch spaces without exhausting the contained solutions inside.

Given a network with incapacitated arcs and no congestion effect, assigning the traffic flow between any pair of nodes to its shortest path minimizes the objective function value. In other words, an all-or-nothing traffic assignment is consistent with the system minimization objective. This simple fact results in the following monotonicity property: if network N_1 includes all the arcs of network N_2 , i.e., $A_1 \supseteq A_2$, where A_1 is the arc set of N_1 and A_2 is the arc set of N_2 , then,

$$c(\mathbf{x}_1, \mathbf{z}_1) \leq c(\mathbf{x}_2, \mathbf{z}_2), \quad (2.26)$$

because adding a link to an existing network never lengthens a shortest path between any pair of nodes. By applying this monotonicity property, the lower bound of any branch space can be estimated by computing the objective function value of a network including all the arcs in its undetermined arc set S_U :

$$c(\mathbf{x}_L, \mathbf{z}_L): S_I^L = S_I \cup S_U, S_E^L = S_E, \text{ and } S_U^L = \emptyset. \quad (2.27)$$

Under the same branching framework, Hoang (1973) and Dionne and Florian (1979) suggested an improved lower bound that can greatly accelerate the above branch-and-bound search. Hoang (1973) first found that a tighter bound can be derived based on the following fact: given $A_1 \supseteq A_2$, where A_1 and A_2 represent the arc sets of two networks N_1 and N_2 respectively, we have

$$c(\mathbf{x}_1, \mathbf{z}_1) + \sum_{a \in A_1 - A_2} x_a \Delta c_a \leq c(\mathbf{x}_2, \mathbf{z}_2), \quad (2.28)$$

where Δc_a is the travel cost increment for a unit trip between the two end nodes of arc a , as caused by deleting arc a from network A_1 . Apparently, the left-hand side of inequality (2.28) provides a stronger bound to the right-hand side than that of inequality (2.26). Based on Hoang's work, Dionne and Florian (1979) further improved the algorithm by specifying a better arc selection order that is always choose the arc with the highest $\Delta c_a/d_a$ value for branching. With this improved bounding rule, the lower bound for a current branch space is written as, given that the current space is spanned by the included arc set S_I , excluded arc set S_E and undetermined arc set S_U ,

$$c(\mathbf{x}_L, \mathbf{z}_L) + \sum_{a \in S_U} x_a \Delta c_a: S_I^L = S_I \cup S_U, S_E^L = S_E, \text{ and } S_U^L = \emptyset. \quad (2.29)$$

The lowered bound developed by Hoang (1973) and Dionne and Florian (1979) is applicable to the network design problem with the operation cost minimization objective subject to the capital cost constraint (see (2.1)-(2.8)), which is the co-called budget design problem. Los and Lardinois (1982) extended their idea to the problem case of minimizing the sum of the network operation cost and capital cost (see (2.9)-(2.15)). Given the following inequality of capital cost between two networks N_1 and N_2 with $A_1 \supseteq A_2$, where A_1 and A_2 are the arc sets of N_1 and N_2 , respectively, that is,

$$d(\mathbf{z}_1) \geq d(\mathbf{z}_2) + \sum_{a \in A_1 - A_2} d_a z_a, \quad (2.30)$$

the lower bound for the mixed-cost objective function of a current network solution can be readily obtained by combining inequalities (2.28) and (2.30), such as,

$$c(\mathbf{x}_L, \mathbf{z}_L) + d(\mathbf{z}_K) + \sum_{a \in S_U} (x_a \Delta c_a + d_a z_a): S_I^L = S_I \cup S_U, \quad (2.31)$$

$$S_E^L = S_E, \text{ and } S_U^L = \phi; S_I^K = S_I, S_E^K = S_E \cup S_U, \text{ and } S_U^K = \emptyset.$$

In addition to the different lower bounds, a number of different branch-and-bound tree structures were also proposed for discrete network design problems. Ridley (1968), for example, used an alternative tree structure for the incapacitated, non-congested budget design problem. In his search tree, every vertex represents a particular solution, which is simply denoted by two types of arcs, included arcs and excluded arcs. If we still use S_I and S_E to represent the included arc set and excluded arc set respectively, the status of the root vertex in this tree structure can be further written as $S_I = A_F \cup A_V$ and $S_E = \emptyset$. In other words, the network solution at the root vertex is the full network that includes all the candidate arcs. This full network, however, is often an infeasible or non-optimal solution since it requires the maximum capital cost. Starting from the root vertex, each branch will be generated by selecting one candidate arc from S_I to S_E . Unlike the first tree structure in which each branching process always produces two branches that correspond to selecting one arc from S_U to S_I and from S_U to S_E respectively, this branching scheme produces a different number of branches at a different vertex, where the number is dependent on the remaining number of arcs that are not yet selected from S_I to S_E as well as the budget constraint. Suppose that n branches are generated by a branching process. Then the branching operation can be written as,

$$S_I^1 = S_I - \{a_1\} \text{ and } S_E^1 = S_E \cup \{a_1\}, \quad (2.32)$$

$$S_I^2 = S_I - \{a_2\} \text{ and } S_E^2 = S_E \cup \{a_2\}, \quad (2.33)$$

.....

$$S_I^n = S_I - \{a_n\} \text{ and } S_E^n = S_E \cup \{a_n\}. \quad (2.34)$$

Although this alternative branching scheme generates a different tree structure, the monotonicity property is still applicable to establishing the bounding rule. It is readily known that the objective function value of the solution at any vertex is a lower bound to all its successors, since the included arc set of any successor solution is always a subset of the included arc set of the solution at the current vertex. Therefore, the key algorithmic operation as the search encounter a vertex is conditional on a decision based on the bounding examination result: if the lower bound is lower than or equal to the upper bound (i.e., the objective function value of the updated best solution), the vertex as well as its successors is eliminated; otherwise, a new branching should be promoted from this vertex.

Scott (1969) used the same branching and bounding rules as well as the tree structure as above in his network optimization procedure. However, the search strategies in these two studies differ in terms of their defined search orders over the tree search space. Ridley's search procedure employs a top-down process that the search starting from the root vertex scans all the vertices in the first generation, which are formed by transferring one arc from S_I to S_E , and then scans the next generation, until the last generation at the bottom. In contrast, Scott's procedure searches and bounds solutions according to a path-by-path order, that is, first tracking one complete path from the root vertex to a terminal vertex and then the second path until all root-terminal paths are scanned. The featured search step of Scott's algorithm is to backtrack to its predecessor vertex for an alternative path whenever the current vertex is proved as a

terminal vertex due to infeasibility or bounding. Because of this algorithmic feature, Scott's algorithm falls into the backtracking strategy.

An analogous algorithm with the similar backtracking scheme and bounding rule to Scott (1969) was used by Poorzahedy and Turnquist (1982) for solving their approximate equilibrium network design problem. This simplified problem formulation that is approximate to the LeBlanc's (1975) model is attractive since it relaxes the bi-level complexity. More importantly, it retains the aforementioned monotonicity property for establishing the bounding rule, that is, adding an arc to an existing network will always decrease or at least not increase the user-equilibrium objective function value.

In Poorzahedy and Turnquist's search tree, as similar to the tree structure used in Ridley (1968) and Scott (1969), every vertex represents a single network solution and the backtracking search scheme is used. However, the different aspect is that their branching scheme starts at the root vertex with a minimum network that only contains the fixed arcs, i.e., $S_I = A_F$ and $S_E = A_V$. A new branch as well as its corresponding vertex is produced by selecting a candidate arc from S_E to S_I . By comparing the two algorithmic designs, it is clear that their search directions are in reverse: Scott's procedure has a search direction from the maximum network $A_F \cup A_V$ to the minimum network A_F , while Poorzahedy and Turnquist's procedure invokes a solution evolution from the minimum network A_F to the maximum network $A_F \cup A_V$.

It may be difficult to make a clear judgment on which branching or backtracking search scheme is superior to another. In fact, selecting an appropriate scheme between the two in terms of the computational efficiency is highly dependent on the specific

structure and features of the target problem. For example, one of the factors affecting this choice is the tightness of the budget constraint. If the allowable budget level is relatively high, the search scheme initialized from the maximum network may be a better choice to exhaust the feasible regions; if the budget level is low, then a search starting from the minimum network may be preferred. An extensive experiment by Dionne and Florian (1979) showed a mixed result in testing and comparing the performance of different bounding and branching choices.

2.2.3.1.2 Benders decomposition

Benders decomposition (Benders, 1962) is an algorithmic approach to solve some mathematical programming problems with complicating variables, in which if these complicating variables are temporarily fixed, the remaining optimization problem becomes considerably more tractable. This feature exists in many mixed integer programming problems, where the integer variables are generally regarded as complicating variables. Thus, it is not surprising that the Benders decomposition framework has been applied early to exploit the structure of discrete network design problems and construct solution strategies for them. Magnanti and Wong (1984) and Magnanti et al. (1986) summarized and developed a number of Benders decomposition applications in solving incapacitated, fixed-cost network design problems, in which the underlying traffic assignment problem is an all-or-nothing assignment problem. The basic Benders decomposition approach was generalized by Geoffrion (1972) to accommodate the nonlinear mixed integer programming problem, so that a large range of flow-dependent variable-cost discrete network design problems could be fit into this decomposition framework.

The Benders decomposition technique decomposes a discrete network design problem into two parts, a discrete part containing the design variables, $z_a, \forall a \in A_V$, and a continuous part containing the flow variables, $x_a, \forall a \in A_C$. The decomposition can be seen from the following rewritten functional formulation to the discrete network design problem, if we use, for example, the first problem formulation described early in this text,

$$\min \quad c(\mathbf{z}) \tag{2.35}$$

s.t. constraints (2.2)-(2.3)

$$\text{where } c(\mathbf{z}) = \min_{\mathbf{x}} \left\{ \sum_{a \in A_C} x_a c_a(x_a) : \text{constraints (2.4)-(2.8)} \right\} \tag{2.36}$$

given that for every design solution \mathbf{z} , there is at least one feasible flow solution \mathbf{x} . (If this assumption does not hold, a similar but more generalized functional formulation should be resulted (refer to Geoffrion (1972)). However, this imposed assumption does not alter the nature of our discussed problem while it reduces the complexity of the formulation.) This new problem formulation can be regarded as the projection of the original problem onto the feasible space of \mathbf{z} .

Note that in general the evaluation of the subproblem $c(\mathbf{z})$ for any specific \mathbf{z} , i.e., the solution of the traffic assignment problem in (2.36), can be efficiently obtained. By the duality theory or Lagrangian relaxation, the optimal solution of this problem can be derived by solving the following equivalent program:

$$c(\mathbf{z}) = \max_{\lambda \geq 0} \left\{ \min_{\mathbf{x}} \left\{ \sum_{a \in A_C} x_a c_a(x_a) - \sum_{a \in A_C} \lambda_a (M_a z_a - x_a) : \text{constraints (2.4)-(2.6)} \right\} \right\} \tag{2.37}$$

where λ_a denotes the dual variable or Lagrangian multiplier.

With this duality device, the original problem consequently collapses to the following decomposition form:

$$\min \quad c(\mathbf{z}) \tag{2.38}$$

s.t. constraints (2.2)-(2.3)

$$c(\mathbf{z}) \geq \min_{\mathbf{x}} \left\{ \sum_{a \in A_c} x_a c_a(x_a) - \sum_{a \in A_c} \lambda_a (M_a z_a - x_a) : \text{constraints (2.4)-(2.6)} \right\} \tag{2.39}$$

which we call the master problem. The natural strategy for solving the master problem is relaxation, since it contains a large number of constraints shown in (2.39). One generally begins by solving a relaxed version of the master problem by ignoring all but a few of these constraints. Meanwhile, the solution of the subproblem is used to check the feasibility of the optimal solution of the relaxed problem to the original problem and to provide cutting planes for further reforming or redefining a tighter master problem. By taking advantage of this decomposition scheme, we may solve the original discrete network design problem by iteratively solving a series of relaxed master problems and subproblems.

One must note that the choice of constraints for relaxation in the subproblem is an algorithmic choice, as dependent on the specific structure of the network design problem as well as other algorithmic elements. The above functional formulations merely provide a general example for illustrating the methodology. Implementations

of the Benders decomposition framework may vary in terms of the algorithmic design and problem structure.

Hoang (1982) first applied the generalized Benders decomposition to solve an equilibrium discrete network design problem. In his setting, the flow conservation constraints (i.e., constraints (2.4) and (2.5)) were dualized. With a set of given values of the discrete design variables as an initial solution, an iterative procedure is used to solve the dual subproblem and the master problem respectively with exchanging the evolving network and flow information between them. Due to the decomposition, the dual subproblem with the fixed values of the design variables and the master problem with the fixed values of the flow variables can be relatively readily solved by the existing algorithms.

Sherali et al. (1991) proposed and solved a discrete network design problem in an integrated evacuation routing and shelter location application. By assuming that there is a central authority managing the evacuation process, they formulated a network design model using a system-optimal objective function and a convex flow-dependent link cost function (i.e., the BPR function), but without appending a user-optimal routing constraint. The discrete decision variable in their model is the choice of shelter locations at the given candidate destination nodes. This model can be fit into the general discrete network design model presented above. To see the connection, let us add a dummy node and a set of dummy candidate links to the existing network, where a candidate link connects a candidate destination node where a shelter could be built. If a candidate node is chosen in a feasible solution, it is equivalent to its connecting candidate link is chosen. In such a way, this discrete location choice problem is converted to be a discrete link choice problem, in which the set of dummy

candidate links becomes the variable link set A_v . Since each shelter has an accommodating capacity, correspondingly, this problem becomes a capacitated network design problem.

In their study, a heuristic algorithm and an exact enumeration algorithm based on the generalized Benders decomposition were used to solve this capacitated nonlinear mixed integer programming problem. In their network design case, Benders decomposition proceeds iteratively by choosing a tentative feasible network, solving for the optimal objective function for this tentative network, and then using the solution to redefine a new network configuration. As similar to Hoang (1982), the tentative network flow problem is a convex minimum cost network flow problem; the authors employed a Lagrangian dual method to solve it. The dualized constraints, however, are the capacity constraints (i.e., constraint (2.7)), as different from Hoang (1982). The optimal solution for a tentative network provides the Benders cuts for the network design problem. In their implementation of Benders decomposition, a strongest surrogate constraint (Parker and Rardin, 1988) was developed to evolve the optimal solution from a continuous relaxation of the discrete network design problem, in which the integer constraint $z_a = 0$ or 1 is relaxed to $0 \leq z_a \leq 1$.

The heuristic and exact algorithmic procedures in Sherali et al. (1991) are different by their mechanisms of implementing the above conceptual approach. The heuristic procedure proceeds by sequentially solving the continuous relaxation and fixing certain discrete choices until all the discrete decision variables are fixed. The exact optimization procedure, like a branch-and-bound algorithm, uses a binary tree structure to search for the optimal solution, by iteratively completing and branching a

vertex, in which the strongest surrogate constraint and general Benders cuts are repeatedly used.

2.2.3.2 Approximate methods

While the exact methods such as the branch and bound and Benders decomposition techniques have been successfully applied to solve some discrete network design problem examples, the computational obstacle generally precludes them from being used in dealing with transportation networks of realistic size. In some cases, when it is difficult to establish effective bounds to eliminate a large number of solution spaces, the exact method tends to be an exhaustive enumeration and it loses its practical value from the computational perspective. Therefore, researchers have early paid attention to another algorithmic paradigm that includes a variety of heuristic methods. The main motivation for developing heuristics for discrete network design problems is to enable those intractable problems to be studied and solved approximately, if not optimally (Pearman, 1974). Computational efficiency and algorithmic tractability are the most important practical advantages of heuristic methods. Heuristics provide a very attractive alternative approach in developing solutions for discrete network design problems.

Existing heuristics for discrete network design problems were designed on the basis of a variety of different mechanisms and assumptions, from an analogous standard optimization technique with a tighter, though less restricted bounding process, to a problem-specific, empirically customized partition or aggregation strategy, from a statistical optimization approach with some assumption about the distribution of (local) optimal solutions, to a set of systematic, iterative solution updating procedure

specified by a metaheuristic framework. We limit our attention in this review to only those that were of specific applicability to the discrete network design problem type we described above, since many heuristics were tailored to exploit the special characteristics of this certain problem type and hence their effectiveness and performance are heavily problem-dependent.

2.2.3.2.1 *Tree search*

As we mentioned above, a possible way to develop good heuristics is to adopt the tree search structure of the branch-and-bound method with some more powerful but less restricted bounds so as to eliminate a larger number of solution spaces and accelerate the search process. Dionne and Florian (1979) first discovered the possibility of using the global travel cost change to take place of the local travel cost change in the bounding inequality, where the latter was originally defined by Hoang (1973). Their new bounding notion can be written as, given $A_1 \supseteq A_2$,

$$c(\mathbf{x}_1, \mathbf{z}_1) + \sum_{a \in A_1 - A_2} \Delta c(\mathbf{x}_{A_1 + a}, \mathbf{z}_{A_1 + a}) \leq c(\mathbf{x}_2, \mathbf{z}_2), \quad (2.40)$$

where $\Delta c(\mathbf{x}_{A_1 + a}, \mathbf{z}_{A_1 + a})$ denotes the total travel cost decrease in the objective function when link a is added into network A_1 . This inequality is conceptually tighter than the one given by Hoang (see (2.29)), which provides a stronger bound for the elimination of non-optimal solution spaces. However, its correctness cannot be guaranteed in general, in that there exists a risk that the optimal solution in a given network design problem be eliminated by the bounding process. Thus, a branch-and-bound procedure based on this bounding principle can only be considered as an approximate method. Nevertheless, the computing experience reported in Dionne and Florian (1979) of

implementing this heuristic seems very encouraging that the overall probability of overlooking optimal solutions was almost ignorable so that the optimal solutions were obtained in most of the problem cases and all the problem solutions were within 0.03% of optimality.

2.2.3.2.2 Linear programming relaxation

As we have seen, the incapacitated, fixed-cost version of discrete network design problems is formulated as a linear mixed integer program. Relaxation of the integer constraints on the design elements results in a linear programming problem, whose optimal solution provides a lower bound to the original network design problem. Balakrishnan et al. (1989) first derived a linear programming relaxation problem from the network design formulation with disaggregate forcing constraints, since this relaxation version in general provides tighter bounds than that from the aggregate forcing constraints and then developed a dual-based technique for the solution of this linear programming relaxation problem. This relaxation strategy provides a few attractive features that can accelerate the solution of the original problem: first, the dual-ascent technique can generate a relatively tight lower bound; second, the dual-ascent technique can be used to identify a feasible network solution used as an initial point for other heuristics; third, it can be used to eliminate those design variables that will be excluded in the optimal solution so as to reduce the solution spaces. The effectiveness of this method was demonstrated by an implementation of the combination of the dual-ascent procedure and a drop-and-add heuristic.

A different linear programming relaxation method for the same problem was proposed and implemented by Lamar et al. (1990). While their strategy is still to seek a lower

bound for the optimal solution through a dual-ascent method, the linear programming relaxation program is resulted from the original problem with aggregate forcing constraints. The power of this bounding rule is enhanced by adjusting the capacity parameter, M_a , so as to produce a lower bound as tight as the one given by the relaxation program from the problem formulation with disaggregate forcing constraints. Its advantage, however, lies in that the resulting relaxed linear program is equivalent to a shortest path problem, which can be easily solved.

2.2.3.2.3 Approximate dynamic programming

A set of recursive heuristic procedures have emerged in the literature for solving discrete network design problems, which imitates the algorithmic structure of the dynamic programming method used to solve some classical combinatorial problem, such as the knapsack problem. While a dynamic programming procedure could be included in a broad range of branch-and-bound processes, it has its own distinct algorithmic characteristics and scope of applicability.

Along with an application of the branch-and-bound method, Scott (1969) proposed two approximate rank-and-select procedures for solving the discrete network design problem with a constrained budget. The first approximate procedure gets started with a minimum spanning tree solution and then proceeds forward by adding links toward a feasible solution with as many as links included, while another procedure, named the backward procedure, initiates from a fully connected network solution and moves backward by deleting links. The shared part of the two procedures is a subroutine used for comparing all the solutions generated by adding (or deleting) a candidate link into (or from) the network during the forward (or backward) process. In Scott's

design, the forward procedure stops when no more link can be added into the network without violating the budget constraint, while the backward procedure, when it is implemented, must work with the forward procedure together, in that the backward procedure stops deleting any link once the budget constraint is satisfied and the forward procedure continues the following search until no more link can be added. While these procedures have a similar recursive process to dynamic programming algorithms, the arbitrary rule of selecting a link to add (or delete) with maximizing the decrease (or minimizing the increase) of the objective function value ignores the combinatorial nature of the problem and makes them merely approximate algorithms. Dionne and Florian (1979) modified Scott's backward algorithm by considering both the travel cost and capital cost changes caused by deleting a link instead of the travel cost only. Numerical results showed that this modification made a considerable improvement on the problem optimality while the computational efficiencies of the two versions of the backward heuristics are comparable.

Another heuristic with the rank-and-select operation was devised by Boffey and Hinxman (1979). Their procedure calls a start from an empty network and continues adding new links into the network according to an ordered link list determined by the information about the changes of local travel cost and capital cost caused by adding each single feasible link. This process stops once the allowed largest number of links in the network by the budget constraint is reached. Now an earliest network solution on the list is backtracked that has not been used as a starting point of any adding process, and then a new ordered link list is generated from this starting point and a new link-adding process is resumed. Such an adding-and-backtracking procedure is repeated until no feasible solution in the current list is available or a pre-specific number of iterations are exhausted.

Janson and Husaini (1987) applied a very similar heuristic method to Boffey and Hinxman's to solve a fixed-cost network design problem and the same problem plus the combined trip distribution and traffic assignment (refer to (2.18)-(2.20)). In both of these procedures, the "benefit-over-cost" ratios were used as the criterion to determine the order of selecting links. However, there are two major discrepancies in Janson and Husaini's algorithmic design: first, their procedure works in a backward fashion, starting from an entire connected network solution, i.e., the solution including all the candidate links, and gradually deleting non-preferred links; second, the calculation of the "benefit-over-cost" ratio, or more precisely speaking, the travel cost change due to deleting (or adding) a link, is based on a global impact rather than a local impact. By considering the global optimality, of course, we would seemingly more confident on this global, network-wide travel cost evaluation in the ranking operation; however, in the meantime, additional computational cost needs to be paid for a larger number of objective function evaluations.

From the perspective of algorithm design, in which the solution quality and computational efficiency are both the critical evaluation factors, there is no rigorous evidence showing which one is preferable to the other between these two methods. In a following study, Janson et al. (1991) extended this approach with incorporating several extra algorithmic elements and adapted it to accommodate a multi-period transportation network design problem. These added operations include, for each iterative operation, allowing multiple top candidate links rather than the only best one to be deleted from or added into the current solution, and allowing a swap between a link in the current solution and a link deleted previously.

2.2.3.2.4 Geometric analysis

Geometric analysis can be often used to provide spatial insights on the problem structure and result in good approximate solutions in many combinatorial problems with Euclidean features. A geometry-based heuristic method was proposed by Wong (1985) for a special Euclidean version of the basic fixed-cost, budget-constrained network design problem. The special settings include that all the nodes are located in a unit circle, the travel cost and capital cost with any link are the Euclidean distance of this link, and the demand rate between any O-D pair is the unit rate. The main operation of this heuristic is to apply a pre-specified capital cost threshold to filter candidate links to be added into the network. The filtering process is repeated with the continuously decreasing threshold value until the network solution generated at the end of a filtering process is feasible. A network built by connecting the closest node to the circle center with each of other nodes is used as the initial network solution. Though the above procedure was originally designed for the network design problem with the geometric characteristics described above, it may not be difficult to extend it, with some scaling modifications, to be applied to a general budget-constraint network design problem.

2.2.3.2.5 Greedy search

We may have noted that the main algorithmic designs in the heuristics listed above are quite problem-specific. In the combinatorial optimization field, many difficult problems could also be satisfactorily solved under some general-purpose heuristic frameworks—metaheuristics—such as, for example, greedy search, genetic algorithm, scatter search, simulated annealing, tabu search, and nested partition, to name a few.

As we identified in our survey, the value of metaheuristics for solving discrete network design problems has been justified in a number of recent case studies. Greedy search is an optimization technique that belongs to the family of local search. A typical greedy search procedure starts from an initial solution and applies a systematic rule to search for and move to the best solution in the neighborhood until no improvement can be made by the rule. It is often described as a “hill-climbing” strategy, in that the process of moving step by step along a trajectory to a local optimum mimics the path to a hill top. The advantage of this technique applied to a discrete network design problem is that the highly complex functions and constraints can be relatively readily handled as a drop-and-add process at each algorithmic move.

Billheimer and Gary (1973) devised a greedy search method for the incapacitated, non-congested network design problem, whose main algorithmic steps include the repeated use of a link elimination routine and an insertion routine. Along these link operation routines, a bounding process is also established to reduce the number of solutions to be evaluated in the neighborhood. Los and Lardinois (1982) applied a similar local search strategy with some improved algorithmic elements to solve the same network design problem. To enhance the global optimality or, in other words, to increase the probability of local optima being the global optimal solution, their local search procedure was repeated by introducing multiple randomly generated initial solutions through a statistical optimization method.

2.2.3.2.6 Genetic algorithm

Genetic algorithms are one of stochastic global optimization techniques. The basic idea of genetic algorithms was inspired by Darwin’s theory of evolution. Its

algorithmic procedure imitates the natural selection and survival of the fittest in the evolution of species. In the literature, there have been a few discrete network design studies carried out with the use of a genetic algorithm, including Xiong and Schneider (1992, 1995), Jeon et al. (2006) and Meng et al. (2008). The target problem in all these studies is of the bi-level network design form. A genetic algorithm can be described as a generation-to-generation evolution process, in which any generation is formed by a set of chromosomes (solutions). Following a population of randomly generated initial individuals, each offspring in the new generation is produced by some basic operations such as crossover (i.e., recombination of two or more solutions), mutation (i.e., random variation of a solution), and reproduction (i.e., born a new child solution with some appropriate stochastic rules), in terms of the knowledge from its ancestors, which, of course, are typically the most useful knowledge. These basic operations are often tailored to incorporate some problem-specific attributes or properties so that a better understanding about the problem structure could be absorbed in the generational process. A typical genetic algorithm requires two things to be defined: a genetic representation of the solution domain and a fitness function to evaluate the solution domain. A termination condition is typically used to determine when the generational process should stop.

2.2.3.2.7 Simulated annealing

Another metaheuristic that of the stochastic global optimization features is simulated annealing, the name and principle of which are inspired by annealing in metallurgy, a statistical mechanics technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects. For a typical simulated annealing procedure, there are three control parameters: the starting temperature, the

number of iterations, and the factor by which the temperature is reduced at each iteration. At each step of a simulated annealing process, the current solution is replaced by a random neighboring solution, chosen with a probability that depends on the difference between the corresponding function values and on the global temperature, which is gradually decreased during the process. The dependency is such that the current solution changes almost randomly when the temperature is high, but increasingly “downhill” as the temperature goes to the zero level. The allowance for “uphill” moves prevents the method from being stuck at local minima. Simulated annealing has been applied to solve both discrete and continuous network design problems. In discrete problem cases, for example, Drezner and Wesolowsky (1997, 2003) and Cantarella et al. (2006) respectively applied a simulated annealing procedure to tackle a special discrete network design problem, in which the increasing problem complexity arises from the lane-based capacity allocation that each link to be designed includes the following allowable decisions: no-built scheme, two-way scheme, and two one-way schemes.

2.2.3.2.8 Tabu search

Tabu search, belonging to the class of local search techniques, is another metaheuristic strategy that has been employed to solve discrete network design problems. As similar to other local search methods, a tabu search process searches the neighborhood of the current solution and moves to the best solution in the neighborhood. Such a search process iteratively continues until some stopping criterion is satisfied. Tabu search enhances the performance of a local search method by using memory structures, in which the oscillatory phenomenon of repeatedly visiting previous solutions on the search itinerary can be avoided by marking a recently visited solution as a tabu and

excluding it from the allowable solution set. To mitigate the excessive problem in assigning tabus, an aspiration criterion is in general suggested so as to override the otherwise excluded solutions. Intensification and diversification rules are also often used, in order to tailor the search direction and distribute computational efforts in different search regions. Mouskos (1991) first investigated the effectiveness and efficiency of a tabu search method in finding optimal network designs for a set of user-equilibrium network design problems. His computational results showed that tabu search is an efficient method in finding high-quality solutions.

It has been realized that a capacitated network design problem is considerably more difficult to solve than an incapacitated one. Crainic et al. (2000) successfully devised a simplex-based tabu search procedure for lessening the algorithmic challenges arising from the capacitated problem. The unique feature of their procedure is that an integrated rather than separated functional form is employed of discrete decision variables and continuous flow variables in the algorithmic design. On the basis of a path-based formulation of the capacitated network design problem, the procedure combines simplex-based pivot moves with column generation to yield a search that explores the space of the continuous path flow variables, while evaluating the actual mixed integer objective of the original problem. An important contribution from this study is that it demonstrates an effective method of combining the search power of a metaheuristic with the structural analysis capability from linear programming.

2.2.3.2.9 Partition, aggregation and reduction

Researchers also sought ways to approximate the discrete network design problems themselves instead of the solutions, so as to reduce the computational complexity to

the level that can be accommodated by the standard exact algorithms. Burstall (1966) employed a network partition strategy to decompose an electricity power supply network design problem into a series of local subproblems that were assumed as approximately spatially independent and then solve these local optimization problems separately. An approximate optimal solution is then obtained by combining the solutions of all the subproblems.

Chan (1976) and Chan et al. (1989) developed a three-stage process to solve large transportation network design problems, in which a heuristic procedure first transfers the original problem to be an abstract problem with controllable size by categorizing and aggregating network nodes and links, then a branch-and-backtrack algorithm is employed to solve the abstract problem, and finally the abstract optimal solution is translated back to the original disaggregate network. An alternative aggregation strategy through extraction is proposed by Haghani and Daskin (1984, 1986), in which minor links and nodes are deleted from the original network and travel demand rates in the reduced network are adjusted accordingly.

The introduction of the user-equilibrium constraint into a transportation network design problem is behaviorally desirable but adds extra modeling complexity and computational burden. Though the standard branch-and-bound algorithm was successfully tailored for the optimal solution of this particular network design case (see LeBlanc, 1975), the lower bound formed by the use of the system-optimal traffic assignment in general is rather loose, which means that a relatively long enumeration process should be expected. A formulation-approximation approach suggested by Poorzahedy and Turnquist (1982), as we introduced above, relaxed the required computational cost to a large extent. The replace of a user-equilibrium objective

function to the original system-optimal function reduces the bi-level discrete network design problem to a single level.

In addition to these problem-approximation techniques, continuous network design problems with capacity expansion (or reduction) are often regarded as approximate versions of the corresponding discrete problems. Although, of course, there are many network control and management problems (e.g., traffic signal control and ramp meter operation) to which the continuous network design models are particularly applicable, relaxing a discrete network design problem to its corresponding continuous version can greatly reduce the computational complexity. An example of this type of relaxation can be seen in Steenbrink (1974b). However, the applicability and extent of using such a relaxation in solving a realistic network design problem may be still questionable.

2.2.3.2.10 Other heuristic procedures

Expert systems and artificial intelligence techniques have also been introduced into solving discrete network design problems as alternative heuristic approaches. For instance, Tung and Scheider (1987) developed a knowledge-based expert system to handle a large-scale, multi-objective discrete network design problem, in which a knowledge base, which contains the specific heuristic rules for solving the target problem, was developed by a human-machine interaction method. Xiong and Schneider (1992) and Wei and Schonfeld (1993), with realizing the computational bottleneck to most of discrete network design algorithms is the evaluation of the objective function, especially to those congestion-dependent networks, suggested

using neural networks to approximate objective function values during the search process so that a large part of computing cost could be saved.

2.2.4 Summary

The synthesis of discrete network design models and solution methods presented above provides us with many useful modeling insights and algorithmic ingredients in the model formulation and solution development. Two of them are summarized below.

On the modeling side, we want to emphasize the importance of the model formulation and selection in modeling a discrete network design problem. Formulating a proper optimization model, to some extent, is also an optimization process subject to multiple factors and requirements, such as the purpose and scale of the target problem, model applicability and dimensionality, modeling rationale and precision, individual behavior assumption, as well as the available computing resources. On the one hand, we need to make the model reflect the basic structural relationship among all the modeling components and capture the synthetic behavior of the system; on the other hand, we must manage to represent the model in a tractable mathematical form and control the problem-solving cost in an acceptable range. In many cases, the model formulation and solution development should be considered simultaneously. In other words, we need to maintain a trade-off between the model complexity and tractability at the model formulation stage.

On the solution side, we want to clarify that the algorithms and heuristics presented above are not distinctly different methods from each other; instead, many of these solution methods share similar algorithmic principles and components. In fact, some

methods could be jointly used to produce a more efficient and effective method. For example, to give a few, an approximate solution obtained by a heuristic can be often used as a good initial solution for a standard optimization-based method; a dual-ascent method may be used to accelerate the elimination of non-dominant solution spaces in the application of another optimization-based technique; many local drop-and-add algorithmic rules are often incorporated into a metaheuristic procedure. Multiple metaheuristic methods could also be used jointly in a so-called hybrid metaheuristic framework. An instance of hybrid metaheuristics is to combine simulated annealing and tabu search, in which simulated annealing provides a stochastic search framework while tabu search is used to avoid the cycling risk and escape local optima.

Discrete network design problems have been extended to include multiple design dimensions and various design components. With these extensions, the problem complexity and the number of decision variables and solution spaces are increased to an unprecedented level and accordingly pose a new challenge in the solution development. The following chapters discuss the formulation, solution and application of such a complex network design problem encountered in evacuation planning.

CHAPTER 3

MODEL FORMULATIONS

All models are wrong, but some are useful.
—George Box

This chapter describes an optimal evacuation network model and an integrated evacuation network optimization and emergency vehicle assignment model that can be used for evacuation planning in urban traffic networks. The two models are actually related to each other, where the integrated model can be obtained by combining the optimal evacuation network model and an emergency vehicle routing model. Therefore, many fundamental assumptions and settings in our modeling framework are shared by the two models. In the following, we first present these common parts of the two models, including the network representation and notations, travel behavior characteristics and system objective settings, and then describe the modeling rationale and functional forms of the models in detail.

3.1 Network representation and notation

An evacuation network can be denoted by a directed graph, $G = (N, A)$, where N and A are the sets of nodes and links in the graph, respectively. To model the lane-based network details, each intersection and roadway section in the network are represented by an intersection subnetwork and a roadway-section subnetwork, respectively. We call such a graph an *expanded network* that consists of intersection subnetworks and roadway subnetworks. Accordingly, the link set A in an expanded network consists of two exclusive parts: intersection link set A_I and roadway-section link set A_R , i.e.,

$A = A_I \cup A_R$ and $\emptyset = A_I \cap A_R$. As we will specify in details later, links in a roadway subnetwork have different properties from those in an intersection subnetwork under our network settings: a roadway link is treated as an ordinary graphical link, associated with capacity, cost, and other travel supply-demand attributes, while an intersection link is an impedance-free link and only functions with providing the network connectivity.

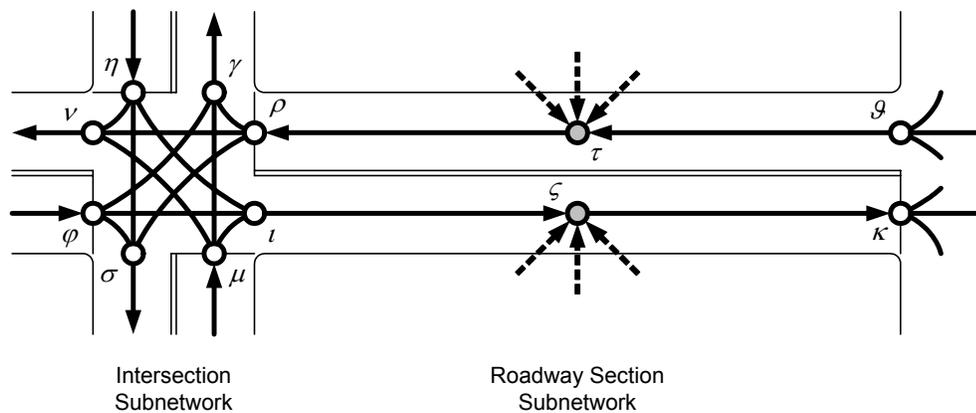


Figure 3.1 Intersection subnetwork and roadway-section subnetwork

An intersection subnetwork and a roadway section subnetwork are illustrated in [Figure 3.1](#). The intersection subnetwork consists of 8 nodes and 12 links (if all the legs of the intersection are two-way roadways and all of the through and turning movements are allowable). The roadway section subnetwork between the two intersections includes 6 nodes and 4 links, where each of the lane directions is represented by a pair of consecutive directional links and for each traffic direction there are one upstream node, downstream node and intermediate node. The upstream and downstream nodes, e.g., nodes ι , κ , ϑ and ρ , provides connections between the roadway section and its adjacent intersections. The intermediate nodes are assigned as traffic source nodes, e.g., nodes ζ and τ . For the modeling convenience, it is arbitrarily assumed that all

the traffic collected by a roadway section from the proximal street block originates from its intermediate node.

The notations used in the models are listed below. Note that both link-based and path-based flow variables are used. From [Figure 3.1](#), we know that there are two types of links in this network representation: roadway-section links and intersection links. As we will see below, the two sets of links in an evacuation network share some fundamental graphical characteristics but have different capacity and connectivity properties. In the following notations, unless otherwise specified, link $\eta \rightarrow \iota$ can be referred to as either a roadway section link or an intersection link.

Parameters and sets:

- $c_{\iota\varsigma}$ single lane capacity of link $\iota \rightarrow \varsigma$, where $\iota \rightarrow \varsigma$ is a roadway section link
- $u_{\eta\iota}$ capacity of link $\eta \rightarrow \iota$, where $\eta \rightarrow \iota$ is an intersection link
- $d_{\eta\iota}$ length of link $\eta \rightarrow \iota$, where $\eta \rightarrow \iota$ may be a roadway section link or an intersection link
- $t_{\eta\iota}^0$ free-flow travel time on link $\eta \rightarrow \iota$, where $\eta \rightarrow \iota$ is a roadway section link or an intersection link
- b_ν net flow rate at node ν
- $n_{\iota\kappa, \vartheta\rho}$ total number of lanes of two adjacent link pairs $\iota \rightarrow \varsigma \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$ in a roadway section between two intersections
- S_ι set of the starting nodes of links pointing to node ι
- R_ρ set of the ending nodes of links emanating from node ρ

Variables:

- $n_{\eta\iota}$ number of lanes on link $\eta \rightarrow \iota$, where $\eta \rightarrow \iota$ is an intersection link or a roadway section link
- $n'_{\eta\iota}$ number of reserved lanes for emergency vehicle use on link $\eta \rightarrow \iota$
- $x_{\eta\iota}$ evacuation flow rate on link $\eta \rightarrow \iota$
- $x'_{\eta\iota}$ emergency vehicular flow rate on link $\eta \rightarrow \iota$ ($x'_{\eta\iota} = 0$ or 1)
- $t_{\eta\iota}$ travel time on link $\eta \rightarrow \iota$, where $t_{\eta\iota}$ is a function of $x_{\eta\iota}$ and $n_{\eta\iota}$
- $y_{\eta\iota}$ connectivity indicator of link $\eta \rightarrow \iota$, where $\eta \rightarrow \iota$ is an intersection link ($y_{\eta\iota} = 0$ or 1)
- $z_{\iota\kappa}$ connectivity indicator of link pair $\iota \rightarrow \mathcal{S} \rightarrow \kappa$, where $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ is a pair of consecutive roadway links ($z_{\iota\kappa} = 0$ or 1)
- q_{rs} evacuation flow rate between origin-destination (O-D) pair $r-s$
- f_k^{rs} evacuation flow rate on path k between O-D node pair $r-s$
- t_k^{rs} true travel time on path k between O-D pair $r-s$
- T_k^{rs} perceived traffic time on path k between O-D pair $r-s$
- ξ_k^{rs} travel time perception error on path k between O-D pair $r-s$
- $\delta_{\eta\iota,k}^{rs}$ path-link incidence indicator denoting the relationship between link $\eta \rightarrow \iota$ and path k ($\delta_{\eta\iota,k}^{rs} = 0$ or 1)

The two connectivity indicators, $y_{\eta\iota}$ and $z_{\iota\kappa}$, are 0-1 dummy variables. When $y_{\eta\iota} = 1$, it indicates that a positive flow on intersection link $\eta \rightarrow \iota$ is allowed; when $y_{\eta\iota} = 0$, it indicates that link $\eta \rightarrow \iota$ is blocked and accordingly no flow is on link $\eta \rightarrow \iota$. When $z_{\iota\kappa} = 1$ (or $n_{\iota\kappa} \geq 1$), it indicates that at least one lane along a pair of consecutive roadway links, $\iota \rightarrow \mathcal{S} \rightarrow \kappa$, is used; when $z_{\iota\kappa} = 0$ (or $n_{\iota\kappa} = 0$), it indicates that $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ vanishes (i.e., the lanes originally designed for this traffic direction are all

reversed for the contraflow direction). The path-link incidence indicator $\delta_{\eta l, k}^{rs}$ determines the relationship between link $\eta \rightarrow l$ and path k between O-D pair $r-s$. If $\delta_{\eta l, k}^{rs} = 1$, it means that link $\eta \rightarrow l$ is on path k ; if $\delta_{\eta l, k}^{rs} = 0$, otherwise. The network configuration decision variables, n_{lk} and z_{lk} , and $n_{\eta l}$ and $y_{\eta l}$, specify the schemes of lane reversal and crossing elimination, respectively.

3.2 Travel behavior

It is critical for an evacuation planning model to properly specify individuals' travel choice behaviors under emergency conditions. An emergency occurrence is such a unique, one-time event that evacuees have much more uncertainty in choosing destinations and routes than in their daily commuting travels. In some cases, there are a set of prepared public refuges by the emergency management authority, while in many other cases, evacuees are only prompted to leave the emergency area as soon as possible. Many evacuees may not choose a specific refuge or place outside the evacuation network as their destination before setting out their evacuating trips. Even if some evacuees have or are given a destination, they may not be able to choose the fastest routes, to get to the destination, because of the lack of day-to-day driving experiences within the evacuation traffic under emergency situations. Numerous experiences showed that evacuees tend to select their evacuating routes and destinations based on their own perceptions of danger and observations of ongoing traffic conditions (Golding and Kasperson, 1988).

Instead of trying to assume that evacuating behaviors could be “fit” into rational, engineered notions, an evacuation plan should be built bottom-up from what is known about actual behavior. We realize that many evacuees in an emergency area would

make their travel choices with some basic knowledge of network connectivity, limited information from an evacuation order announced by the authority, and their own perceptions of the emergency situations. Any rational evacuees would aim to leave the hazardous area as soon as possible. A behavior implication consistent with this fact is that an evacuee often chooses a perceived fastest route to leave the hazardous area first without considering a specific evacuation destination. Accordingly, an evacuation egress would not be selected or perceived by an evacuee prior to his route choice. A more reasonable conjecture about the evacuating population's travel choice behavior is that they determine destination and route choices simultaneously.

Without resorting to complex modeling mechanisms, an integrated destination and route choice concept can be readily modeled by adding a super dummy destination node to the original evacuation network and connect each exit node to this super destination by a dummy link with zero travel impedance. With this one-destination network representation, we do not need to explicitly consider the destination choice modeling in an evacuation network optimization model; instead, an exit node would be determined as a virtual destination, when one chooses a route to the super destination. In other words, a destination choice is implied in the corresponding route choice under this one-destination network setting.

The integrated route and destination choice of an evacuee should be made in terms of the prevailing traffic conditions over the evacuation network. Without precisely knowing about the traffic information over the network, however, he or she may not properly choose a route and destination to minimize his or her own evacuation time. A more reasonable routing behavior is that given a transportation network with or without a publicly announced evacuation plan, any evacuee would choose an

evacuating route based on his own travel time perception. Individual perceived route travel times in general are with some perception errors and are defined as random variables over the evacuating population. Therefore, a stochastic user-optimal flow assignment may be the best in describing an evacuation process with individual perception errors.

In the case of an integrated evacuation network optimization and emergency vehicle assignment, the emergency vehicle fleet typically consists of only a limited number of vehicles (compared to the number of vehicles in the evacuation population) and is assigned with one or a few reserved routes. As a result, the routing behavior for these emergency vehicles is tightly limited (to the reserved route) and no serious congestion effect along this route needs to be considered. The route selection for the emergency vehicle assignment can be simply based on free-flow travel times.

Since the emergency vehicles are generally dispatched from one or more emergency centers outside the disaster area, there is an origin choice issue associated with its route choice. That is, in an evacuation network, which node on the boundary of the network should be chosen as the entry point for an individual emergency vehicle? This entry point is defined as the origin node for the emergency vehicle assignment in the evacuation network. Similar to the integrated destination and route choice modeling mechanism with the evacuation flow modeling, an integrated origin and route choice setting can be applied to the emergency vehicle assignment. By adding a super dummy origin node to the evacuation network and connecting each candidate entry point to this super origin node by a zero-travel-cost dummy link, the origin choice can be completed simultaneously with the route choice for the emergency vehicle assignment.

3.3 System objectives

Network clearance time is one of the most important efficiency indicators that are used to evaluate the performance of an emergency evacuation plan. The objective of an evacuation planning model may be set as to minimize the total travel times of all evacuees or to minimize the maximum of individual evacuation times. The latter is positively related to the network clearance time. The two objectives can be written as

$$\min \sum_{\eta l} x_{\eta l} \cdot t_{\eta l}(x_{\eta l}, n_{\eta l}), \quad (3.1)$$

where η and l are indexes of the end nodes of link $\eta \rightarrow l$, and $x_{\eta l}$, $t_{\eta l}$ and $n_{\eta l}$ are the traffic flow rate, travel time, and number of lanes of link $\eta \rightarrow l$, and

$$\min \max_{k,rs} (t_k^{rs}(\mathbf{x}_k^{rs}, \mathbf{n}_k^{rs})), \quad (3.2)$$

where r and s are the indexes of an origin and exit node, respectively, k is the index of a path between O-D node pair $r-s$, and c_k^{rs} is the travel time of path k between nodes r and s , which is the function of the traffic flow and number of lanes of all the links on path k , i.e., \mathbf{x}_k^{rs} and \mathbf{n}_k^{rs} .

The objective selection for an evacuation network optimization model depends on the duration from the time of disseminating an evacuation order to the expected occurrence time of the hazardous event. The allowable time for a safe evacuation varies in terms of the nature and intensity of the hazardous event. To clarify the

evacuation outcomes from different objective functions, a simple example consisting of two evacuation planning scenarios is illustrated in Figure 3.2. The evacuation performance of each scenario is represented by a cumulative evacuation curve, which depicts the relationship between the cumulative number of evacuated people and the time[†]. It is readily observed that the area between the vertical axis and a cumulative evacuation curve denotes the total evacuation time, while the time moment of the upper end of a curve indicates the maximum of individual evacuation times.

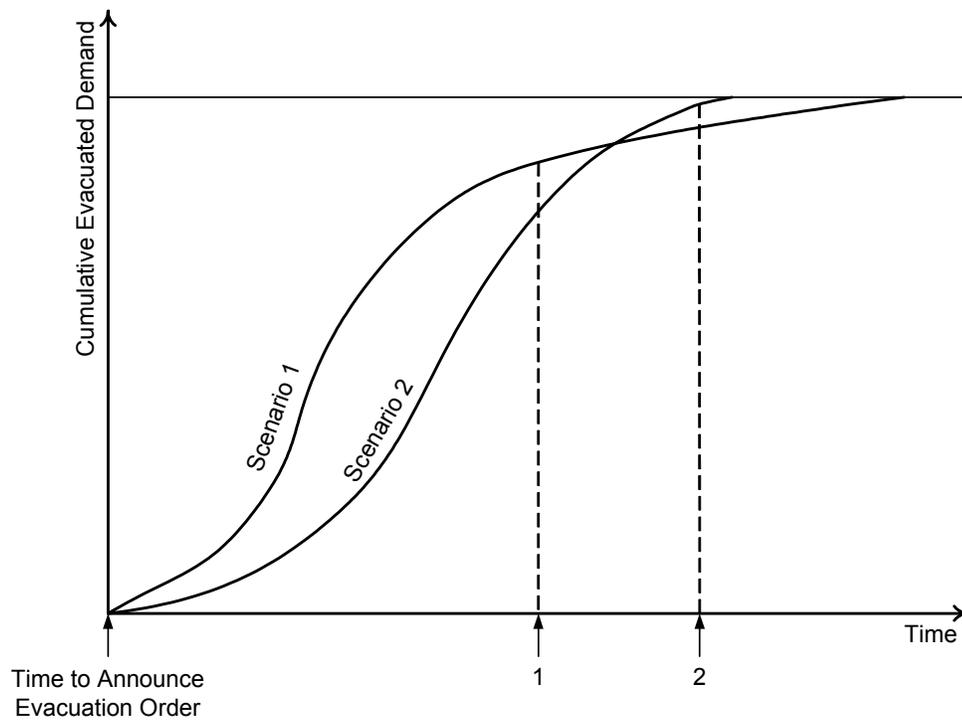


Figure 3.2 A performance comparison of minimizing total evacuation time and minimizing the maximum of individual evacuation times

[†] In general, such an evacuation-versus-time curve reflects the real evacuation performance only in a dynamic evacuation process. We use it here merely as an illustration for the purpose of evaluating planning scenarios.

In this illustration, it is assumed that the two cumulative evacuation curves are the results respectively of using the objectives of minimizing the total evacuation time (scenario 1) and minimizing the maximum of individual evacuation times (scenario 2). The two curves must have (at least) a crossing point. The time moment of this crossing point is critical, indicating that by this time the two planning scenarios have evacuated the same number of people from the hazardous area. Let us then compare the evacuation efficiency of the two scenarios given the expected time of a hazardous occurrence. If the hazardous event is expected to occur before the critical time, e.g., at time point 1, we may believe that scenario 1 is better since it has evacuated more people than scenario 2 before the hazardous event; if the expected occurrence time is at time point 2, the same reason can be applied to conclude that scenario 2 is a better evacuation plan.

Previous evacuation studies showed that the demand departure rates of many evacuation cases have such a temporal pattern that, after the evacuation order is broadcasted, the initial departure rate is relatively low in a short preparation period (say, 30-60 minutes), and then the departure rate significantly increases and quickly reaches its surge level as well as this surge departure rate lasts for a rather long period, until the most of the evacuees (say, 90 percent) leave the emergency area. As an illustration, example cumulative demand generation curves retrieved from several nuclear power station evacuations in the northeastern U.S. (including Vermont Yankee in Vermont, Pilgrim in Massachusetts, and Nine Mile Point and Indian Point in New York) are presented in [Figure 3.3](#). All these cases clearly show a similar temporal demand generation pattern, in which the peak departure rate dominates the whole evacuation period, typically spanning from 40-60 minutes to 2-3 hours. In fact, the major part of the evacuating demand is generated at this peak departure rate, say

around 80 to 90 percent. The most important feature is that the departure rate is relatively stable during the peak period.

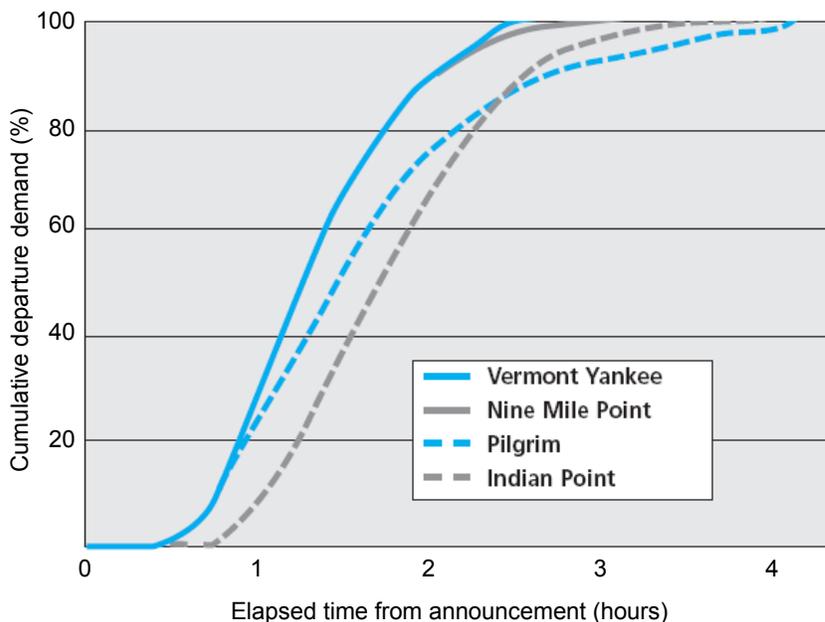


Figure 3.3 Temporal evacuation generation distributions[†]

Therefore, it is reasonable to focus on the evacuation network optimization for the surge level of the evacuation demand rate, as an approximation to the optimal network solution applicable for the whole evacuation process. Given the stable travel demand pattern during the surge period, a static network optimization model can satisfy the requirements of modeling rationale and precision. Without considering the time-dependent demand variation explicitly, the objective of minimizing the total evacuation time is preferred to minimizing the maximum of individual evacuation times for an evacuation planning problem, which is a more appropriate and direct surrogate of the network evacuation performance measure.

[†] Source: Goldblatt and Weinisch (2005).

Compared to the objective of the evacuation network optimization problem, the logic of the emergency vehicle routing is relatively simple, in that its typical task is to find a fastest route from the outside emergency area to a designated inside site (or an area) and reserve a certain number of lanes along this route for its exclusive use, where the number of lanes depends on how many emergency vehicles are dispatched. The number of emergency vehicles may range from several tens to hundreds, depending on the emergency nature and scale. This number, however, is small compared to the whole evacuation demand. Given both the evacuation network optimization and emergency vehicle assignment objectives, the latter often receives the top priority, since, as we described previously, those emergency management officials and technical experts may have the most urgent tasks during an emergency period.

The prior treatment of emergency vehicle routing inevitably imposes extra restrictions on the network capacity and connectivity and hence affects the evacuation network performance. Along the reserved emergency vehicle route, the remaining capacity for evacuating traffic is reduced accordingly. At the intersections passed by an emergency vehicle route, allowable turning movements for the evacuating traffic are more limited, where the full crossing elimination requirement needs to be maintained subject to additional emergency vehicle turning movements[†]. In view of these interactions between the different objectives, the objective of minimizing the total evacuation time should be pursued subject to the prior emergency vehicle routing requirement.

[†] In the case that the size of emergency vehicle fleet is relatively small, we may not need to strictly maintain a crossing-conflict prevention constraint between the evacuating traffic and the emergency vehicles. With the assistance of traffic management personnel at an intersection, the evacuating traffic can be temporarily stopped so as to allow the emergency vehicles to pass, which should not introduce considerable traffic delays to the evacuating traffic if the number of stopping times is minimal.

3.4 An evacuation network optimization model

We propose a bi-level network design model for the evacuation network optimization problem, since the stochastic traffic flow pattern needs to be specified by an equivalent mathematical program (Sheffi, 1985) in the lower level and the objective of this program is different from the system performance measure given by the upper level. The upper-level objective function is,

$$\min z(\mathbf{x}, \mathbf{n}) = \sum_{\eta l} x_{\eta l} \cdot t_{\eta l}(x_{\eta l}, n_{\eta l}) \quad (3.2)$$

where the travel time of link $\eta \rightarrow l$, $t_{\eta l}(\cdot)$, is a function of the link flow, $x_{\eta l}$, and the number of available lanes, $n_{\eta l}$. For an evacuation traffic network, the widely used Bureau of Public Roads (BPR) function is suggested to calculate the link travel time,

$$t_{\eta l}(x_{\eta l}, n_{\eta l}) = t_{\eta l}^0 \cdot \left[1 + \alpha \left(\frac{x_{\eta l}}{c_{\eta l} \cdot n_{\eta l}} \right)^\beta \right], \quad (3.3)$$

where α and β are both link-specific parameters. As we specified previously, the link performance function is only applied to the links in roadway-section subnetworks.

The travel time with an intersection link is assumed to be zero.

There are capacity constraints for roadway links, which confines the total capacity of a roadway section to be shared by its two reverse traffic directions. Given the fixed lane capacity, the capacity exchange between the two directions of the roadway section can be represented by the numbers of their lanes. Refer to the example roadway subnetwork in [Figure 3.1](#), the capacity constraints are written as,

$$n_{\iota\kappa} + n_{\vartheta\rho} = n_{\iota\kappa, \vartheta\rho}, \quad (3.4)$$

$$n_{\iota\kappa}, n_{\vartheta\rho} \geq 0 \text{ and integral}, \quad (3.5)$$

where $n_{\iota\kappa, \vartheta\rho}$ is the total number of lanes of the two directions (i.e., $\iota \rightarrow \zeta \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$) of the roadway section. In the case of a roadway section partially occupied by the emergency vehicle route, constraint (3.4) should be written as $n_{\iota\kappa} + n_{\vartheta\rho} = n_{\iota\kappa, \vartheta\rho} - n'_{\iota\kappa} - n'_{\vartheta\rho}$, where $n'_{\iota\kappa}$ (or $n'_{\vartheta\rho}$) indicates the number of the lanes reserved for the emergency vehicle use on link $\iota \rightarrow \zeta \rightarrow \kappa$ (or link $\vartheta \rightarrow \tau \rightarrow \rho$). This set of constraints regulates the lane reversal configuration.

We also have a set of crossing-elimination constraints for links in the intersection subnetwork. By referring to [Figure 3.1](#), these constraints have two forms, namely, two-link constraints and three-link constraints, as follows,

$$y_{\varphi\iota} + y_{\mu\gamma} \leq 1, \quad (3.6)$$

$$y_{\eta\iota} + y_{\rho\sigma} + y_{\mu\gamma} \leq 1, \text{ and} \quad (3.7)$$

$$y_{\varphi\iota}, y_{\mu\gamma}, y_{\eta\iota}, y_{\rho\sigma} = 0 \text{ or } 1, \quad (3.8)$$

where the first two constraints show that a potential crossing conflict between links $\varphi \rightarrow \iota$ and $\mu \rightarrow \gamma$ and between any two of links $\eta \rightarrow \iota$, $\rho \rightarrow \sigma$ and $\mu \rightarrow \gamma$, respectively, is not allowed in our evacuation plan. In other words, by either of the above constraints, it means that at most one link can carry a positive traffic flow and other links crossing this one must be disallowed. At a four-leg intersection, there exist 16

potential crossing-conflict points, which leads to 4 two-link constraints and 4 three-link constraints. In the case that the assigned emergency vehicle route goes through this intersection, for example, the emergency vehicle route along the direction $\mu \rightarrow \gamma$, $y_{\mu\gamma} = 1$ must be incorporated into constraints (3.6) and (3.7) as an input element.

In the meantime, an inherent relationship between a roadway link connectivity indicator and its corresponding number of lanes needs to be maintained: for example, given z_{ik} and n_{ik} in Figure 3.1, if $z_{ik} = 1$, then $n_{ik} \geq 1$, and vice versa; if $z_{ik} = 0$, then $n_{ik} = 0$, and vice versa. Thus, the following set of inequalities is used to describe this relationship:

$$z_{ik} \leq n_{ik}, \text{ and} \quad (3.9)$$

$$z_{ik} M \geq n_{ik}, \quad (3.10)$$

where z_{ik} is a 0-1 binary integer, n_{ik} is a non-negative integer, and M is an arbitrary, sufficiently large constant. Actually, M is not necessarily a very large number, as long as $M \geq \max_{ik, \vartheta\rho} (n_{ik, \vartheta\rho})$.

As we mentioned earlier, an evacuation network is assumed to be a stochastic user-equilibrium network, which is specified by the lower-level subproblem. An equivalent program to the stochastic user-equilibrium assignment was suggested by Sheffi and Powell (1982). The objective function of the program has the following functional form:

$$\min \sum_{\eta_l} x_{\eta_l} t_{\eta_l}(x_{\eta_l}, n_{\eta_l}) - \sum_{\eta_l} \int_0^{x_{\eta_l}} t_{\eta_l}(\omega) d\omega - \sum_{rs} q_{rs} E \left[\min_k (T_k^{rs}) \mid \mathbf{t}^{rs}(\mathbf{x}) \right] \quad (3.11)$$

The first term of the above objective function is the total travel time (i.e., the function to be minimized in a system-optimal traffic assignment). The second term is the function to be minimized in a deterministic user-equilibrium traffic assignment. In the third term, q_{rs} represents the traffic flow rate between O-D pair $r-s$; the component $E\left[\min_k(T_k^{rs}) \mid \mathbf{t}^{rs}(\mathbf{x})\right]$, $k \in K_{rs}$, where K_{rs} is the set of routes between pair $r-s$, is the expected perceived minimum travel time over all the routes between pair $r-s$. The perceived route travel time, T_k^{rs} , includes two parts, the actual travel time, t_k^{rs} , and the individual perception error, ξ_k^{rs} , i.e., $T_k^{rs} = t_k^{rs} + \xi_k^{rs}$.

In the lower-level subproblem, a set of capacity constraints are obviously required. For consecutive link pair, $l \rightarrow \zeta \rightarrow \kappa$, in the roadway subnetwork in [Figure 3.1](#), the capacity constraints are,

$$x_{l\zeta}, x_{\zeta\kappa} \leq c_{l\kappa} n_{l\kappa}, \text{ and} \quad (3.12)$$

$$x_{l\zeta}, x_{\zeta\kappa} \geq 0, \quad (3.13)$$

and for link $\eta \rightarrow l$ in the intersection subnetwork, the capacity constraints are,

$$x_{\eta l} \leq u_{\eta l} y_{\eta l}, \text{ and} \quad (3.14)$$

$$x_{\eta l} \geq 0. \quad (3.15)$$

Moreover, a set of flow conservation constraints also need to be satisfied. Consider the source nodes (i.e., nodes ζ and τ) in the roadway subnetwork in [Figure 3.1](#).

There are two flow conservation conditions associated with the lane reversal

configuration. When the roadway allows bi-directional traffic (even if one direction is with a reduced number of lanes for the contraflow of its reverse direction), the traffic generated from the origin node on any direction is accommodated by its corresponding traffic lane(s). The flow conservation constraints for origin nodes \mathcal{S} and τ for the two-way traffic operation are respectively,

$$x_{\mathcal{S}\kappa} - x_{\mathcal{S}} = b_{\mathcal{S}}, \text{ and} \quad (3.16a)$$

$$x_{\tau\rho} - x_{\vartheta\tau} = b_{\tau}. \quad (3.16b)$$

On the other hand, when one traffic direction (e.g., $\vartheta \rightarrow \tau \rightarrow \rho$) is fully prohibited for the maximum contraflow capacity on the other direction (i.e., $\iota \rightarrow \mathcal{S} \rightarrow \kappa$), the traffic originating from the origin node (i.e., node s) on this direction will be carried by its reverse direction. It is equivalent to setting the net traffic flow from node τ to be 0 and accordingly increasing the net flow from node \mathcal{S} to $b_{\mathcal{S}} + b_{\tau}$. For this one-way traffic operation, the flow conservation constraints for origin nodes \mathcal{S} and τ are:

$$x_{\mathcal{S}\kappa} - x_{\mathcal{S}} = b_{\mathcal{S}} + b_{\tau}, \text{ and} \quad (3.17a)$$

$$x_{\tau\rho} - x_{\vartheta\tau} = 0. \quad (3.17b)$$

The above two lane operations can be integrated into the following set of flow conservation constraints, with introducing the 0-1 dummy variables $z_{\iota\kappa}$ and $z_{\vartheta\rho}$, which are the connectivity indicators of link pairs $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$ respectively,

$$x_{\mathcal{S}\kappa} - x_{\mathcal{S}} = b_{\mathcal{S}}z_{\iota\kappa} + b_{\tau}(1 - z_{\vartheta\rho}), \text{ and} \quad (3.18a)$$

$$x_{\tau\rho} - x_{\partial\tau} = b_{\tau}z_{\partial\rho} + b_{\varsigma}(1 - z_{i\kappa}). \quad (3.18b)$$

This set of node flow conservation constraints also specifies the O-D demand rates, where source flows b_{ς} and b_{τ} may be transferred between source nodes ς and τ , which is dependent on the values of $z_{i\kappa}$ and $z_{\partial\rho}$.

The flow conservation constraint for other nodes (except for the destination node(s)) has a standard form with the net incoming or outgoing flow equal to 0. For example, for node ι and ρ in [Figure 3.1](#), we have

$$x_{i\varsigma} - \sum_{\eta \in S_i} x_{\eta\iota} = 0, \text{ and} \quad (3.19a)$$

$$x_{\tau\rho} - \sum_{\sigma \in R_{\rho}} x_{\rho\sigma} = 0. \quad (3.19b)$$

For the modeling and solution convenience, the flow conservation constraints (i.e., (3.18) and (3.19)) need to be converted to the O-D flow conservations and are represented by the following constraints denoted by link-based and path-based traffic flow variables,

$$q_{rs} = \sum_k f_k^{rs}, \quad (3.20)$$

$$x_{\eta\iota} = \sum_{rs} \sum_k f_k^{rs} \delta_{\eta\iota, k}^{rs}, \text{ and} \quad (3.21)$$

$$t_k^{rs} = \sum_{\eta\iota} t_{\eta\iota}(x_{\eta\iota}, n_{\eta\iota}) \delta_{\eta\iota, k}^{rs}, \quad (3.22)$$

where these constraints are in a consistent representation form with the objective function of the lower-level subproblem (i.e., (3.11)). In the above constraints, O-D

demand q_{rs} is a function of the node net flows (i.e., b_s and b_τ) and link connectivity indicators (i.e., z_{ik} and $z_{\theta\rho}$), and path-link incidence indicator $\delta_{\eta_i,k}^{rs}$ is determined by the link connectivity indicators.

For the discussion convenience, we refer to the evacuation network optimization model described above the *basic model*.

3.5 An integrated evacuation network optimization and emergency vehicle assignment model

As an extension of the basic evacuation network optimization model, we also considered the following bi-objective evacuation planning problem: minimization of the total evacuation time and minimization of the emergency vehicle routing time.

In general, optimization of a multi-objective system results in a Pareto-optimal set instead of a single optimal solution (or a set of equivalent optimal solutions). A set of solutions are said to be Pareto-optimal if none of these solutions can dominate any other solutions in the set on all the objective measures. In the general multi-objective system paradigm, there have been a number of modeling mechanisms used to simplify the analysis and evaluation of multi-objective systems[‡]. Among them, some widely used multi-objective problem formulation approaches may include:

- *Weighted combination*. When different objective measures are commensurate in quantities, a weighting coefficient can be specified for each individual

[‡] An early summary and annotation of multi-objective transportation systems optimization studies can be found in Current and Min (1986).

objective function and a new objective function results, which is a combination of the products of all objective functions and their respective weighting coefficients. For example, to evaluate the network performance of an urban traffic network, a generalized cost of the traffic system is often defined as a weighted sum of total travel time and monetary cost, where the “value of time” is the weighting coefficient to convert travel time into monetary cost.

- *Constraint surrogate.* The constraint surrogate strategy allows us to turn a multi-objective system into a set of single-objective systems in an alternative way. A single objective is selected as the system objective while other objective functions are surrogated by corresponding constraints given that an acceptable set of values of these objective functions can be specified in advance. An overall evaluation of the multi-objective system will be based on the optimization results of the whole set of single-objective systems. For example, in a multi-objective network design problem, the objective of minimizing design (or construction) costs may be written as an inequality constraint with an upper bound of design budget, while the objective of minimizing user transport costs is kept as the system objective.
- *Lexicographic optimization.* The lexicographic optimization approach is a special case of the more general hierarchical optimization. If the objectives of a system can be clearly ranked in a descending order of importance, the top (most important) objective is first optimized with the relaxation of all other objectives, and then in turn the next objective is optimized subject to an additional constraint that limits feasible solutions to not exceed a pre-specified fraction of the last optimal objective value and so on until all objectives are

exhausted. When the fraction is zero, it reduces to the lexicographic optimization. In a multi-modal transportation system analysis, we may apply the lexicographic optimization method to optimize each modal system in terms of a predetermined priority hierarchy.

There is no universal guideline for the selection of multi-objective problem formulation techniques, though in some cases one technique is apparently preferred to others. For a specific problem, selection of a technique is highly dependent on the problem nature and modeling approach as well as the solution method to be used. Given the modeling rationale and requirements for our bi-objective problem, we suggest a lexicographic optimization problem: the objective of minimizing emergency vehicle routing time is of the more importance while minimizing the total evacuation time is given as the second objective.

The bi-objective lexicographic optimization problem is written in a vector optimization form as follows.

$$\min \begin{bmatrix} z_1(\mathbf{x}') \\ z_2(\mathbf{x}, \mathbf{n}) \end{bmatrix} \quad (3.23)$$

where \mathbf{x}' and \mathbf{x} denote the emergency vehicular flow and the evacuation traffic flow, respectively, and \mathbf{n} represents the set of numbers of lanes on all the roadway sections.

The first objective of minimizing the transportation time of the emergency vehicle assignment simply results in a one-origin, one-destination shortest-path problem with a capacity constraint:

$$\min \quad z_1(\mathbf{x}') = \sum_{\eta_l} x'_{\eta_l} t_{\eta_l}^0 \quad (3.24)$$

$$\text{s.t.} \quad \sum_l x'_{\eta_l} - \sum_l x'_{l\eta} = 1 \quad \eta = e \quad (3.25)$$

$$\sum_l x'_{\eta_l} - \sum_l x'_{l\eta} = -1 \quad \eta = d \quad (3.26)$$

$$\sum_l x'_{\eta_l} - \sum_l x'_{l\eta} = 0 \quad \eta \in N - \{d, e\} \quad (3.27)$$

$$(n'_{\eta_l} - n_r) x'_{\eta_l} \geq 0 \quad (3.28)$$

$$n'_{i_S} \leq n_{\kappa, \vartheta \rho} \quad (3.29)$$

where d and e are the node indexes of the destination to which the emergency vehicles are dispatched and the assumed super origin, respectively, $t_{\eta_l}^0$ is the free-flow travel time of link $\eta \rightarrow l$, $n_{\kappa, \vartheta \rho}$ is the total number of lanes of the two traffic directions (i.e., $l \rightarrow \mathcal{S} \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$) of a roadway section that the emergency vehicle route uses, and n_r is the predetermined minimum number of lanes for the emergency vehicle route. Constraint (3.28) can be interpreted as: if $n'_{\eta_l} \geq n_r$, $x'_{\eta_l} = 0$ or 1; $n'_{\eta_l} < n_r$, $x'_{\eta_l} = 0$. In this shortest-path problem formulation, the emergency vehicular flow is simply reduced to the unit of flow since all emergency vehicles are assigned to a single route and no congestion effect is assumed.

As for the second problem of minimizing the total evacuation time, it has the same formulation as the evacuation network optimization model in [Section 3.4](#):

$$\min \quad z_2(\mathbf{x}, \mathbf{n}) = \sum_{\eta_l} x_{\eta_l} t_{\eta_l}(x_{\eta_l}, n_{\eta_l}) \quad (3.2)$$

$$\text{s.t.} \quad \text{constraints (3.3)-(3.22)}$$

This lexicographic model with the objectives of evacuation network optimization and emergency vehicle assignment is referred to the *extended model* hereafter.

CHAPTER 4

INTEGRATED LAGRANGIAN RELAXATION AND TABU SEARCH

I came, I saw, I conquered.
—Julius Caesar

The optimal network models presented in the last chapter pose some complex combinatorial difficulties. The greatest challenge lies in solving the evacuation network optimization problem with the lane-reversal and crossing-elimination controls. In this chapter, we propose an integrated heuristic solution method to address this network optimization problem.

4.1 Problem complexity

A general algorithmic procedure to tackle combinatorial optimization problems is to start from a feasible solution as an initial point and iteratively update the current solution by a neighborhood search until some pre-specified stopping criterion is met. While it is simple and straightforward to obtain an initial feasible solution in many cases, it may not be an easy task in some others. For our evacuation network optimization problem, some difficulties emerge with implementing such an iterative search method. According to our definition, the original network configuration in a real urban network case, as used for daily commuting traffic, is obviously not a feasible evacuation network solution, because the traffic turning movements at an intersection controlled by traffic signals or stop signs allow many crossing points. An external procedure, if possible, needs to be developed to obtain an initial feasible solution. This adds some extra modeling effort.

Another challenge in applying an iterative search procedure for the network optimization problem is the complexity of defining the neighborhood structure for a local search. An intuitive definition for a candidate move in a neighborhood region may be an arc addition, reduction, or swap (for intersection arcs) and a lane exchange between a couple of arc pairs (for roadway section arcs). It is not hard to speculate that, to satisfy the network connectivity constraints, implementation of a candidate move often requires a set of complex network manipulations. Some complex cases include that, for example, a move gets involved with an intersection arc change, and a move causes a full capacity switch between the two traffic directions of a roadway section, which leads to a complete reversal of one of the directions. Under such situations, extracting an exhaustive candidate list from the neighborhood of a feasible solution becomes a very difficult task.

To overcome the difficulties described above, we propose an integrated Lagrangian relaxation and tabu search (LR-TS) method, which takes advantage of Lagrangian relaxation for problem decomposition and complexity reduction and whose algorithmic design is based on the principles of the tabu search metaheuristic. In the Lagrangian relaxation framework, the set of crossing-elimination constraints (i.e., (3.6)-(3.8)) are relaxed and compensated by a penalty term in the objective function. The relaxed Lagrangian problem is inherently a lane-reversal optimization subproblem plus a penalty term. The evaluation of the penalty term becomes a set of local crossing-elimination optimization subproblems, where each intersection of interest poses one crossing-elimination subproblem.

The rationale behind the application of Lagrangian relaxation comes from the following modeling assumptions. First, note that a general modeling setting used in the proposed network optimization model is that travel costs are all associated with the arcs in roadway subnetworks while intersection subnetworks are merely used for maintaining the network connectivity. In accordance, the intersection crossing-elimination constraints can be regarded as side constraints and the objective function value of the lane-reversal optimization subproblem is actually the system cost with the ignorance of these side constraints. Second, it is expected that the lane-reversal subproblem results in an optimal solution with full lane reversals on a large number of roadway arcs. If this outcome is true, the resulting flow pattern (to the lane-reversal subproblem) may be accommodated locally at many intersection subnetworks without causing any crossing point. Ultimately, as long as we find an optimal solution to the Lagrangian problem with the penalty term value equal to zero, this optimal solution is also optimal to the original network optimization problem. In such a way, the Lagrangian relaxation strategy offers a convenient approach to decompose the problem and hence reduce its structural complexity in that we are able to deal with the lane-reversal subproblem and the crossing-elimination subproblem separately.

The algorithmic search procedure to implement the solution strategy proposed above is elaborated in the following sections. The Lagrangian relaxation framework is first presented where the focus is given to the problem decomposition formulation and the Lagrangian multiplier adjustment mechanism. We then give a detailed description about the algorithmic design of the proposed tabu search procedure, in which the neighborhood structure and local search are defined for a lane-reversal optimization subproblem based on a reduced network. Due to the discrete nature, this search procedure requires an evaluation of the objective function whenever a candidate

feasible solution to the Lagrangian problem needs to be examined. The objective function evaluation can be done by estimating a stochastic traffic flow pattern in the reduced network of the given lane-reversal configuration and to examining the feasibility of the associated crossing-elimination configuration in each intersection subnetwork. Since such an evaluation needs to be conducted frequently, it dominates the computational cost of the whole algorithmic procedure. The major part of the objective function evaluation is a stochastic traffic assignment process. For the efficiency purpose, we employ an analytical algorithm whose efficiency is significantly enhanced by the Clark's approximation method. Meanwhile, we defined the crossing-elimination examination subproblem as another discrete optimization problem, which, as we will see later, is an integer programming problem of relatively small size. The crossing-elimination subproblem can be solved by the classical branch-and-bound algorithm or a simplex-based iterative procedure due to its special structural property.

4.2 Lagrangian relaxation framework

Lagrangian relaxation is a general solution strategy for solving mathematical programs, which permits us to decompose problems to exploit their special structures. It has long been used for discovering theoretical insights and developing solution algorithms for various difficult mathematical programming problems. For many discrete and combinatorial optimization problems, Lagrangian relaxation can be used to relax a set of complicating side constraints to be a penalty term in the objective function (see [Geoffrion, 1974](#), for example). By adjusting the values of Lagrangian multipliers with the penalty term to an appropriate level, we may find the optimal solution by solving

the simplified Lagrangian problem that can often take advantage of various previously developed algorithms.

In our application, by relaxing the crossing-elimination constraints (i.e., Constraints (3.6)-(3.8)) and compensating this relaxation by a penalty term in the objective function, the Lagrangian problem can be written as,

$$\min \sum_{\eta_i} x_{\eta_i} t_{\eta_i}(x_{\eta_i}, n_{\eta_i}) + \sum_{\eta_i, \rho\sigma} P_{\eta_i, \rho\sigma} (y_{\eta_i} + y_{\rho\sigma} - 1)^+ \quad (4.1)$$

s.t. constraints (3.4)-(3.5) and (3.9)-(3.22),

where $(y_{\eta_i} + y_{\rho\sigma} - 1)^+$ represents the maximum function of 0 and $y_{\eta_i} + y_{\rho\sigma} - 1$, i.e.,

$$(y_{\eta_i} + y_{\rho\sigma} - 1)^+ = \max(0, y_{\eta_i} + y_{\rho\sigma} - 1). \quad (4.2)$$

The objective function of this Lagrangian problem consists of two parts, where the first part is the objective function of the original problem, i.e., the total evacuation time, while the second one is the penalty term caused by the Lagrangian relaxation, referring to the sum of all the penalty costs in an evacuation network. In the penalty term, $P_{\eta_i, \rho\sigma}$ is a Lagrangian multiplier ($P_{\eta_i, \rho\sigma} \geq 0$), which, in our case, is also called *unit penalty cost*. This unit penalty cost is used to compensate the violation of a single crossing-elimination constraint $y_{\eta_i} + y_{\rho\sigma} \leq 1$, where y_{η_i} and $y_{\rho\sigma}$ are a pair of intersection arcs that have a potential crossing point (refer to [Figure 3.1](#)).

Note that in the penalty term the use of the maximum form $(y_{\eta_i} + y_{\rho\sigma} - 1)^+$ instead of a general constraint relaxation form $y_{\eta_i} + y_{\rho\sigma} - 1$ as the relaxation surrogate does not

change the bounding principle of the Lagrangian relaxation. As we showed below, it is clear that the Lagrangian problem we constructed above is still the lower bound of the original network optimization problem. The reason we employed this maximum function is, as we will discuss later on, to use it to conveniently count the number of violated crossing-elimination constraints. The bounding principle for this particular Lagrangian relaxation is,

$$\begin{aligned}
& \min \left\{ \sum_{\eta_l} x_{\eta_l} t_{\eta_l}(x_{\eta_l}, n_{\eta_l}) : \text{constants (3.3)-(3.22)} \right\} \\
& = \min \left\{ \sum_{\eta_l} x_{\eta_l} t_{\eta_l}(x_{\eta_l}, n_{\eta_l}) + \sum_{\eta_l, \rho\sigma} P_{\eta_l, \rho\sigma} (y_{\eta_l} + y_{\rho\sigma} - 1)^+ : \text{constants (3.4)-(3.22)} \right\} \quad (4.3) \\
& \geq \min \left\{ \begin{array}{l} \sum_{\eta_l} x_{\eta_l} t_{\eta_l}(x_{\eta_l}, n_{\eta_l}) + \sum_{\eta_l, \rho\sigma} P_{\eta_l, \rho\sigma} (y_{\eta_l} + y_{\rho\sigma} - 1)^+ : \\ \text{constants (3.4)-(3.5), (3.9)-(3.22)} \end{array} \right\}
\end{aligned}$$

An important issue related to the effectiveness of this Lagrangian relaxation method is how to determine the values of those unit penalty costs. Too high penalty costs may result in the tabu search process deviating from the track of minimizing the true objective (i.e., total evacuation time) and possibly block the search process to enter some promising feasible region, while too low penalty costs may entrap the search process into an unfeasible region (to the original problem).

The conventional way to tackle this issue is to employ the subgradient method, which is to adjust the unit penalty costs based on the results of repeatedly solving Lagrangian problems until the unit penalty cost values converge to a satisfied level. The subgradient updating procedure, however, implies running the whole tabu search procedure iteratively and may not be a cost-effective approach in our case.

Another feasible approach of circumventing this task is to integrate a unit penalty cost updating mechanism within the tabu search procedure (Gendreau, 2002). Different from the penalty cost updating mechanism based on optimal Lagrangian problem solutions as in the subgradient method, the use of iteration-based self-adjusting penalty costs is much more efficient and flexible. At any iteration point in a tabu search process, the evacuation network solution is examined for the existence of any crossing-elimination violation and the relevant result is recorded into a frequency-based memory, which will then be used to make adjustments to the current unit penalty cost values (i.e., the coefficient values of the penalty term or the Lagrangian multipliers). As we will show later, this examination is equivalent to checking whether the value of each penalty cost component is zero. If a crossing-elimination constraint is frequently violated (i.e., $y_{\eta} + y_{\rho\sigma} - 1 > 0$), its corresponding unit penalty cost should be increased; otherwise (i.e., $y_{\eta} + y_{\rho\sigma} - 1 \leq 0$), its unit penalty cost should be decreased. Some heuristic rules need to be developed for this penalty cost adjustment manipulation in terms of the constraint violation frequency. With the continuously updated unit penalty costs, it is expected to find an optimal (or near optimal) solution to the original network optimization problem by solving the Lagrangian problem. Such a penalty self-adjusting mechanism embedded in a tabu search procedure was successfully implemented in Gendreau et al. (1994).

In the Lagrangian relaxation framework, the problem constraints reduce to the set of constraints (3.4)-(3.5) and (3.9)-(3.22), which confine the lane-reversal configuration. In accordance with this relaxation, an intersection subnetwork in its expanded topology is reduced to a node, since the sole lane-reversal manipulation does not take into account the crossing-elimination configurations at intersections. We call the

graphical topology with this reduction of intersection subnetworks a *reduced network*. As an illustration, the resulting reduced version from the original network in Figure 3.1 is shown in Figure 4.1, where the intersection subnetwork with 8 nodes and 12 arcs is replaced by a single node. It is obvious that the network reduction greatly reduces the complexity of the network topology.

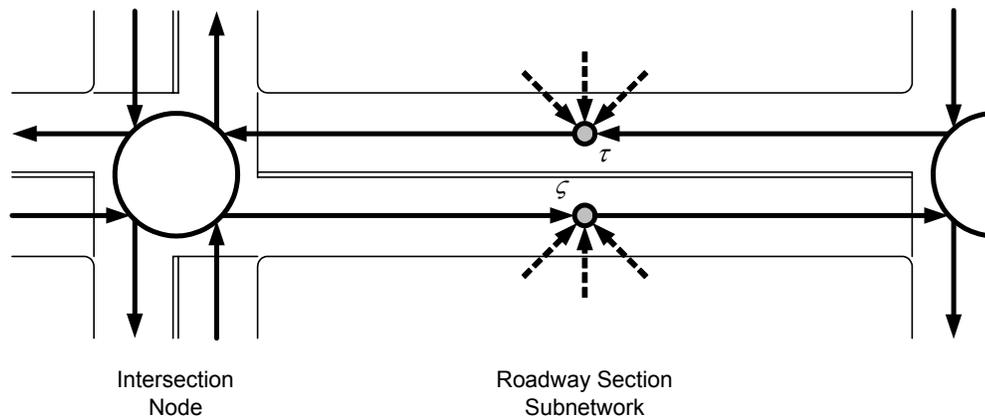


Figure 4.1 Intersection subnetwork reduction

As we alluded to earlier, an existing network solution without any lane-reversal and crossing-elimination configuration (e.g., the existing traffic network configuration in the real world for the daily commuting traffic) can be used as an initial solution for the Lagrangian problem. At the initial phase of the integrated search procedure, due to the lack of information about the iterative intersection crossing-elimination violation behaviors, we may conveniently set all unit penalty costs equal to zero, which means that the search procedure actually starts with a pure lane-reversal network optimization problem without considering the penalty or delay caused by traffic crossing at intersections.

4.3 Tabu search metaheuristic

The relaxed Lagrangian problem is still a complex combinatorial optimization problem with bi-level structure. To tackle its combinatorial complexity, we propose below a tabu search procedure to search for the optimal solution to the Lagrangian problem and update the Lagrangian multipliers.

Tabu search is one of the metaheuristic optimization techniques that are usually used to guide and orient the search of other (local) search procedures. The foundation of tabu search is generally attributed to Glover (1986), in which he described the present form of this technique we use today. Though it belongs to the class of local search techniques, tabu search enhances the performance of a local search method by using memory structures. Tabu search uses a neighborhood search procedure to iteratively move from a solution to another neighboring solution, until some stopping criterion is satisfied. To explore regions of the search space that would be left unexplored by the local search procedure and escape local optimality, tabu search modifies the neighborhood structure of each solution as the search progresses. The solutions admitted to the new neighborhood are determined through the use of special memory structures.

As a metaheuristic optimization technique, tabu search is more of a general problem-solving strategy and optimization framework than any single solution method. There is no universal algorithmic procedure of tabu search that works for all types of combinatorial optimization problems. The general paradigm of tabu search needs to be implemented separately for each application, with the search space, neighborhood structure and other subordinate heuristic components that are specially designed for

the target problem. In the following, we present the algorithmic choices and detail the procedure steps used to address our evacuation network optimization problem with the lane-reversal and crossing-elimination configurations.

Our tabu search method follows a rather straightforward manner: starting with a feasible lane-reversal network solution, the search proceeds with a sequence of local searches and diversification phases until a predetermined stopping criterion is met. Each local search scans all candidate lane-reversal network solutions in the neighborhood with evaluating the objective function of the Lagrangian problem for each solution. In addition to a network flow estimation process, the objective function evaluation also includes a set of intersection crossing-elimination feasibility examinations. An iteration is finished by accepting the best network solution in the neighborhood as the new current solution. Such a scanning and selection process finally stops with the best feasible solution (to the original network optimization problem) encountered during the search once a predefined number of diversification phases has been performed. The whole search procedure can be schematically written as:

Step 0. Choose an initial solution s in the search space S . Set $s^* = s$, $i = 0$ and $j = 0$.

Step 1. Set $i = i + 1$ and conduct a diversification move.

Step 2. Set $j = j + 1$ and generate a subset S^* of candidate solutions in the neighborhood $N(s)$ of s in terms of the recency-based memory and aspiration criterion.

Step 3. Choose an elite subset S^{**} in S^* , and conduct local moves belonging to S^{**} as well as update the recency-based and frequency-based memories, aspiration criterion, current solution s and best solution s^* .

Step 4. If a stopping criterion for the local search is met, go to the next step; otherwise, go to step 3.

Step 5. If a stopping criterion for the diversification search is met, stop; otherwise, go to step 1.

4.3.1 Search neighborhoods and moves

There are two types of neighborhoods in the proposed tabu search procedure: an *adjacent neighborhood* for the local search phase, and a *distant neighborhood* for the diversification search. This section focuses on adjacent neighborhoods and local moves. A distant neighborhood is used to guide the diversification search to enter an unvisited feasible region and its neighborhood structure and moving mechanism is distinct from an adjacent neighborhood, which will be discussed in a later section concerning longer-term memories.

The adjacent neighborhood for a current lane-based network solution is made up of all the lane-based network configurations that can be reached by a single lane-reversal transformation from the current solution. The capacity exchange with such a lane-reversal operation only occurs between the two reverse traffic directions of a roadway section. In other words, a move is defined as a lane exchange between the two arc directions in a roadway section subnetwork. A single lane reversal may change the capacity of the relevant arcs only or change both the capacity and connectivity of the network, depending on the number of lanes to be reversed and the number of lanes on these two reverse directions before and after the lane reversal. Due to the discrete nature of the lane-reversal configuration, we may define three types of moves to reach a candidate solution in the neighborhood.

For a potential move only involving the capacity exchange but not modifying the network connectivity, its lane-reversal direction may be ideally determined by comparing the marginal costs to the whole network generated by the two potential directions with regard to reversing a unit capacity. The lane-reversal direction with the negative marginal cost should be accordingly selected. However, given the discrete requirement, the capacity-reversing amount is quantified by the number of lanes, which in general does not necessarily match the appropriate amount demanded by the desired lane-reversal direction. On the other hand, an accurate estimation of the marginal cost to the whole network with regard to a capacity exchange must be evaluated in terms of some sensitivity analysis technique. Given the stochastic user-equilibrium traffic flow pattern in the evacuation network, the sensitivity analysis is complicated. Because of these reasons, we resort to an approximation method to determine the lane-reversal direction that is to compare the marginal costs with the two potential lane-reversal directions to the local roadway subnetwork.

Let us refer to [Figure 3.1](#), where it is assumed that in the roadway subnetwork all the roadway arcs, including arcs $l \rightarrow \mathcal{S}$, $\mathcal{S} \rightarrow \kappa$, $\vartheta \rightarrow \tau$ and $\tau \rightarrow \rho$, have the same free-flow travel times and lane capacities, as notated by $t_{\eta l, \vartheta \rho}^0$ and $c_{\eta l, \vartheta \rho}$, and the BPR function is used to describe the volume-delay relationship of these arcs with the same arc-specific parameters, α and β (see (3.3)). By assuming that a unit capacity exchange between the two traffic directions in this local roadway subnetwork does not considerably change the traffic flow pattern in the whole network, the local marginal cost with regard to transferring a unit capacity to arc pair $l \rightarrow \mathcal{S} \rightarrow \kappa$ from its counter arc pair is, given the capacity reservation constraint $n_{\kappa} + n_{\vartheta \rho} = n_{\kappa, \vartheta \rho}$,

$$\begin{aligned}
\frac{\partial T_{\iota\kappa, \vartheta\rho}}{\partial n_{\iota\kappa}} &= \frac{\partial [x_{\iota\mathcal{S}} t_{\iota\mathcal{S}}(x_{\iota\mathcal{S}}, n_{\iota\kappa}) + x_{\mathcal{S}\kappa} t_{\mathcal{S}\kappa}(x_{\mathcal{S}\kappa}, n_{\iota\kappa}) + x_{\vartheta\tau} t_{\vartheta\tau}(x_{\vartheta\tau}, n_{\vartheta\rho}) + x_{\tau\rho} t_{\tau\rho}(x_{\tau\rho}, n_{\vartheta\rho})]}{\partial n_{\iota\kappa}} \\
&= \frac{t_{\iota\kappa, \vartheta\rho}^0 \alpha}{c_{\iota\kappa, \vartheta\rho}^\beta} \left[\frac{\partial (x_{\iota\mathcal{S}}^{\beta+1}/n_{\iota\kappa}^\beta + x_{\mathcal{S}\kappa}^{\beta+1}/n_{\iota\kappa}^\beta)}{\partial n_{\iota\kappa}} + \frac{\partial (x_{\vartheta\tau}^{\beta+1}/n_{\vartheta\rho}^\beta + x_{\tau\rho}^{\beta+1}/n_{\vartheta\rho}^\beta)}{\partial n_{\vartheta\rho}} \frac{\partial n_{\vartheta\rho}}{\partial n_{\iota\kappa}} \right] \quad (4.4) \\
&= t_{\iota\kappa, \vartheta\rho}^0 c_{\iota\kappa, \vartheta\rho}^\beta \alpha \beta \left[- \left(\frac{x_{\iota\mathcal{S}}}{c_{\iota\kappa}} \right)^{\beta+1} - \left(\frac{x_{\mathcal{S}\kappa}}{c_{\iota\kappa}} \right)^{\beta+1} + \left(\frac{x_{\vartheta\tau}}{c_{\vartheta\rho}} \right)^{\beta+1} + \left(\frac{x_{\tau\rho}}{c_{\vartheta\rho}} \right)^{\beta+1} \right],
\end{aligned}$$

where $c_{\iota\kappa}$ and $c_{\vartheta\rho}$ are the capacities of arc pairs $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$, respectively, and $c_{\iota\kappa} = c_{\iota\kappa, \vartheta\rho} n_{\iota\kappa}$ and $c_{\vartheta\rho} = c_{\iota\kappa, \vartheta\rho} n_{\vartheta\rho}$.

The marginal cost to the local roadway network with regard to increasing a unit capacity to arc pair $\vartheta \rightarrow \tau \rightarrow \rho$ is known as equivalent to the local marginal cost with regard to decreasing a unit capacity from arc pair $\iota \rightarrow \mathcal{S} \rightarrow \kappa$, that is,

$$\frac{\partial T_{\iota\kappa, \vartheta\rho}}{\partial n_{\vartheta\rho}} = t_{\iota\kappa, \vartheta\rho}^0 c_{\iota\kappa, \vartheta\rho}^\beta \alpha \beta \left[\left(\frac{x_{\iota\mathcal{S}}}{c_{\iota\kappa}} \right)^{\beta+1} + \left(\frac{x_{\mathcal{S}\kappa}}{c_{\iota\kappa}} \right)^{\beta+1} - \left(\frac{x_{\vartheta\tau}}{c_{\vartheta\rho}} \right)^{\beta+1} - \left(\frac{x_{\tau\rho}}{c_{\vartheta\rho}} \right)^{\beta+1} \right]. \quad (4.5)$$

Note that $x_{\iota\mathcal{S}}/c_{\iota\kappa}$, for example, is the volume-over-capacity (V/C) ratio of arc $\iota \rightarrow \mathcal{S}$, which indicates the congestion level of this arc. In view of a pair of arcs on each traffic direction, we further define a congestion measure for these arc pairs. Here, for two arc pairs $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$, $\mathbf{g}_{\iota\kappa} = (x_{\iota\mathcal{S}}/c_{\iota\kappa})^{\beta+1} + (x_{\mathcal{S}\kappa}/c_{\iota\kappa})^{\beta+1}$ and $\mathbf{g}_{\vartheta\rho} = (x_{\vartheta\tau}/c_{\vartheta\rho})^{\beta+1} + (x_{\tau\rho}/c_{\vartheta\rho})^{\beta+1}$ are their congestion measures, respectively. The determination of the lane-reversal direction can then be reduced to a comparison of the congestion conditions of the two traffic directions. That is, when arc pair $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ is more congested than arc pair $\vartheta \rightarrow \tau \rightarrow \rho$ in terms of their congestion measure values, i.e., $\mathbf{g}_{\iota\kappa} > \mathbf{g}_{\vartheta\rho}$, a candidate move should be chosen with reversing a lane from arc pair

$\vartheta \rightarrow \tau \rightarrow \rho$ to arc pair $l \rightarrow \mathcal{S} \rightarrow \kappa$; when $g_{l\kappa} < g_{\vartheta\rho}$, a candidate move should be chosen by the reverse direction; when $g_{l\kappa} = g_{\vartheta\rho}$, no lane reversal is required.

After determining the lane-reversal direction by comparing the congestion measures in a roadway subnetwork, we need to further choose the capacity-reversing amount for the move. Here we employ a simple lane-reversal operation: reverse one lane from the relatively uncongested traffic direction to its counter direction. This arbitrary reverse-one-lane-at-a-time policy seems conservative, but it may be the safest choice in view of two reasons. First, given the combinatorial nature of the problem we have no way to know the appropriate capacity amount or the appropriate number of lanes for exchange between a couple of reverse traffic directions. Second, the lane-reversal direction with a move is determined in terms of the local marginal cost calculation, behind which the presumable rationale is that the network flow pattern does not change considerably with a move. A too drastic move may cause a significant change of the network flow pattern and hence violate the assumption on which the selection of lane-reversal direction is based.

Given the lane-reversal direction and amount specified by the above analysis, the first type of moves is defined as follows: to transfer one lane from the relatively uncongested traffic direction to the congested direction if there are one or more lanes along both of the traffic directions in a given roadway subnetwork. It is applicable to the case in which both of the traffic directions in a roadway section convey a significant amount of traffic flow.

The second type of move is more drastic in changing the lane-reversal network configuration in that these moves get involved in a network topology modification.

More specifically, it reduces the number of arcs in the network by reversing all the lanes along a traffic direction in an eligible roadway subnetwork. Let us suppose the following network configuration and traffic conditions in the roadway subnetwork shown in [Figure 3.1](#): there is one or more lanes on both of the traffic directions, i.e., $n_{l\kappa}, n_{\vartheta\rho} \geq 1$, but the traffic flow on arc pair $l \rightarrow \varsigma \rightarrow \kappa$ is equal or close to zero (which also implies that the traffic demand from the source node ς , b_ς , is equal to or close to zero). A straightforward response to this situation is that the capacity of arc pair $l \rightarrow \varsigma \rightarrow \kappa$ is fully or extremely underutilized and hence all of its lanes should be fully reversed to serve the traffic flow on the reverse arc pair $\vartheta \rightarrow \tau \rightarrow \rho$. This observation defines the second type of moves: a full reversal of lanes along a traffic direction should be given when there is no (or nearly zero) traffic along this direction.

One should note that, with the arc reduction caused by a move of the second type, it is possible, although the possibility is very low in a real traffic network, to form a network with some source nodes unserviceable. Therefore, following the identification of a move of the second type, it is necessary to conduct a feasibility test on the network connectivity. A move that yields an infeasible network configuration should not be considered as a candidate solution. Please also note that for a move of the first type, if the uncongested traffic direction has only one lane, it actually collapses to a move of the second type, in that such a move not only exchanges the capacity between the two traffic directions but also reduces the number of valid arcs in the network.

The third type of move is a rather simple case, which may arise following an iteration that implemented a move of the second type. In a given network configuration, in case that all of the lanes in a roadway section are used to serve only one traffic

direction (e.g., $n_{\iota\kappa} = 0$ and $n_{\vartheta\rho} = n_{\iota\kappa, \vartheta\rho}$ in [Figure 3.1](#)), no matter how congested these arcs are with the current traffic direction, a candidate move is suggested that a lane should be deducted from its current direction for contraflow. Here, for simplicity, the reverse-one-lane-at-a-time policy is used once again.

In many urban traffic networks, in fact, such one-direction roadway sections exist in downtown areas, for some traffic control and safety reasons. The main purpose of designing one-way streets is to decrease the number of the potential traffic crossing points at their connecting intersections and hence reduce the control delays and create a safer driving environment. In the application of our heuristic procedure for an evacuation network optimization problem, if many one-way streets exist in the initial network configuration and/or a large number of full lane reversals emerge in the search itinerary, this type of moves would be frequently encountered. In contrast to the second type, a move of the third type adds an arc pair into the network.

The conditions and manipulations of all the three types of adjacent neighborhood moves can be summarized as follows. Given a roadway section subnetwork as the one in [Figure 3.1](#), we can determine the lane-reversal direction as well as the number of reversed lanes in terms of the if-then rules given below (see [Figure 4.2](#)). Here let us suppose that we are concerned about arc pair $\iota \rightarrow \mathcal{S} \rightarrow \kappa$, as an example, where $g_{\iota\kappa}$ and $g_{\vartheta\rho}$ are the congestion measures for arc pairs $\iota \rightarrow \mathcal{S} \rightarrow \kappa$ and $\vartheta \rightarrow \tau \rightarrow \rho$, as defined previously, and $n_{\iota\kappa, \vartheta\rho}$ is the total number of lanes in this roadway section.

As for the distant neighborhood, it is used to guide the diversification search to enter an unexplored feasible region and accordingly modify the network configuration in a drastic manner. Since the neighborhood structure and moving mechanism of the

distant neighborhood is distinct from the adjacent neighborhood, its implementation will be discussed in another section concerning longer-term memories.

```

if  $n_{ik} = 0$  then
     $n_{ik} = 1$  and  $n_{\vartheta\rho} = n_{ik, \vartheta\rho} - 1$ ;           // 3rd type of move
elseif  $n_{ik} > 0$  and  $g_{ik} = 0$  (or  $g_{ik} \approx 0$ ) then
     $n_{ik} = 0$  and  $n_{\vartheta\rho} = n_{ik, \vartheta\rho}$ ;           // 2nd type of move
elseif  $g_{ik} > g_{\vartheta\rho}$  then
     $n_{ik} = n_{ik} + 1$  and  $n_{\vartheta\rho} = n_{\vartheta\rho} - 1$ ;   // 1st type of move
end;
end;
end;

```

Figure 4.2 A local move candidate selection procedure

4.3.2 Elite candidate list

In a local neighborhood search, a scan will exhaust all the eligible roadway subnetworks with the current network solution and choose a candidate move from each roadway subnetwork into a candidate list. In classic tabu search applications, a single best move will be selected from this candidate list, in terms of the objective function evaluation results as well as subject to the current tabu list and aspiration criterion. This move is then implemented to generate an updated network solution. It is doubtless that this best-candidate-only policy provides a precise ordering of most descent moves, but it may not sufficiently exploit the value of the candidate list, which is determined each time by an exhaustive evaluation of all the eligible lane-reversal

operations in the whole network and whose identification process is the most time-consuming computational part of the whole search process in our case.

A more efficient method is to select and implement a set of moves in a batch after a candidate list is determined. In our case, a promising move set may be the one corresponding to a series of similar or compatible lane-reversal operations along a major roadway route (e.g., a freeway or arterial corridor) or a set of moves that can always reduce the objective function value significantly in a broad range of network connectivity and capacity states close to the current one. This technique is motivated by the assumption that a critically good move, if not performed at the present iteration, will still be a good move for a number of following iterations.

It may be difficult to identify a best set of moves from a given candidate list in a straightforward manner, which, obviously, poses another combinatorial optimization problem. We suggest a simple heuristic rule here to select a set of moves that may better take advantage of the information contained by a candidate list and accelerate search iterations. An *elite candidate list* is elected from the candidate list, which consists of a given number of best candidate moves based on the sorting result of their resulting objective function values. The size of this elite candidate list, where we name it *elite capacity*, is an algorithmic parameter, which indicates how far at most a search can move or how many moves at most a search can convey each time after a move candidate list is presented. If we set the elite capacity equal to 1, it reduces to the classic best-candidate-only policy. An appropriate elite capacity value should be given so as to choose those apparently promising moves in a move candidate list and maintain a good trade-off between the solution quality and search efficiency. If the elite capacity is too large, the fidelity of choosing those moves located in the rear part

of the elite list may not be guaranteed; if the elite capacity is too small, it may not adequately take advantage of the results from sorting the candidate moves and hence the search procedure still frequently demands the highly intensive computation task that is to select and evaluate moves in the candidate list exhaustively.

One must note that with this elite candidate list strategy, even if a precise ordering of implementing the best move at each iteration may not be maintained, it does not necessarily mean that the search quality will be sacrificed. On the one hand, in many cases, the final effect of implementing a set of moves simultaneously would be the same as that of implementing these moves sequentially, though the orderings of the moves in these two different implementations may be different. On the other hand, an accelerated search driven by this more drastic move mechanism may let us conduct more iterations and explore more feasible regions in a given amount of time. The overall performance of a tabu search algorithm should be based on the quality of its best solution and the time spent in finding this solution.

In our case, the moves in an elite candidate list will be evaluated and executed individually, in a consecutive manner, instead of an aggregate form. That is to say, after an elite list is identified, a repeated evaluation of the objective function is conducted each time a move from the elite list is implemented. The implementation of a move is actually determined by this repeated evaluation. Such a sequential move evaluation and implementation mechanism depicts the complete search itinerary. The information recording search iterations provides a direct aspiration criterion and is stored in longer-term memories for the subsequent use of the intensification and diversification strategies. More importantly, this complementary objective function evaluation at each iteration can identify the true contribution of a move to the

objective function value and hence be used to determine its final qualification for implementation. If the performance of a move shown in the candidate list is not consistent with its actual performance in the search itinerary (e.g., a move decreases the objective function value in the preliminary evaluation process for the elite move selection but increases the objective function value in the complementary evaluation), the implementation of this move should be disallowed.

4.3.3 Tabu list and aspiration criterion

The *tabu list* and *aspiration criterion* may be the two most important and essential components in a tabu search heuristic. These two memory-based elements are used to record various information (e.g., solution values and attributes) of the solutions encountered in a search history. The recorded pieces of information are favor of exploring the solution space and guiding the search direction.

The purpose of using a tabu list is to avoid cycling traps and hence escape local optima in a tabu search process. A tabu list typically contains a set of most recent moves, which is constructed based on the concept of recency-based memory.

Whenever a candidate move is identified during the search process, it is compared to the recorded members in the tabu list. If the comparison tells that a member in the tabu list is exactly the counter operation of the candidate move, this candidate move is labeled as a tabu and its candidacy will be canceled unless it satisfies the aspiration criterion. Since it is based on a recency-based memory, a new member is included and the oldest member is discarded each time a move is implemented to the current solution. The general updating mechanism for a tabu list is to put the latest implemented move into the tabu list in place of the oldest member.

In our lane-reversal optimization subproblem, since any lane exchange caused by a move occurs merely between the two directions of a roadway section, we do not need to record both the participating arcs where one obtains capacity while the other loses. Instead, a more effective tabu-recording rule is to add the arc that obtains capacity at the current iteration into the tabu list. At each iteration, a tabu examination invokes a comparison between the arc that potentially obtains capacity through a candidate move and all the members in the tabu list. If the comparison indicates that this arc is equivalent to a member in the current tabu list (i.e., it has the same arc index and the same number of lanes as a tabu member), this candidate move under consideration is regarded as a tabu move and should be accordingly prohibited in the immediate iteration.

A critical parameter associated with a tabu list is the size of the list, which in general is termed as *tabu tenure*. We need to finely tune this parameter with the purpose of preventing local cycling occurrences while not blocking potential promising moves during a search process. A general good range of tabu tenures is about 5-12 (Glover, 1990). However, effective tabu tenures are heavily related to the specific type of target problem instances and have been empirically shown to depend on the size of problems (Glover and Laguna, 1997). Moreover, tabu tenure may be a static value or a randomly or systematically dynamic variable confined by a range. For a particular class of applications, its value or its value-varying mechanism should be calibrated by some empirical calibration process.

One should note that the tabu recording mechanism described above is a simplified version of recording a set of complete solutions visited at the last iterations. While it

occupies less memory, the use of moves (instead of solutions) in the tabu list might lead us to improperly impose the tabu status to solutions that have not been actually visited before. To remedy this information loss and relax the resulting redundant tabu restriction, an aspiration criterion is often used so as to avoid overriding the tabu status to recently conducted moves. While a number of different forms of aspiration criteria have been designed to enable a tabu search heuristic to achieve its high performance level, we employ its most primitive form—the objective function value—to potentially remove a tabu status otherwise applied to a move, because of its wide acceptance with the good performance and its ease of use in a variety of applications. In our implementation, if a move results in a solution whose objective function value is improved compared to the best solution known in the current search history, this move should be permitted even if it has been identified as a tabu move.

Since we take the objective function evaluation as the aspiration criterion, computing the objective function value is inevitable for evaluating any candidate move, no matter if it belongs to a tabu or not. Also, because of this reason, the sequence of the tabu test and aspiration test can be actually reversed in a tabu search procedure.

4.3.4 Intensification and diversification

Both the *intensification* and *diversification* strategies in a tabu search procedure typically resort to the use of longer-term memories. For these longer-term strategies, the modified or enhanced neighborhoods often contain solutions that are identified as elements of a regional cluster in intensification phases and as elements of different clusters in diversification phases. Though the use of longer-term memories is optional, a tabu search procedure enhanced by longer-term strategies can often find very high-

quality solutions within a more economical time span than that only using short-term memories.

Longer-term memories are frequency based. We use a frequency measure termed *residence frequency* to evaluate the need for intensification and diversification, which is defined as the ratio of the number of iterations where an attribute or element belongs to solutions in a search itinerary (or a section of this itinerary) over the total number of iterations in this itinerary (or the corresponding section). The purpose of using residence frequency is to keep track of how often attributes or elements are members of the historical solutions or how frequently in the search history they satisfy some specific status.

In our lane-reversal optimization subproblem, intensification is useful when a roadway subnetwork is set for a specific lane reversal on a very frequent level, which indicates that a move representing an alternative lane configuration of this subnetwork is seldom selected into the elite candidate list. Such an event occurs, for example, when one traffic direction on a roadway section is barely used in a variety of solutions and adding a lane to this direction would cause a large increase to the total travel time in the network. We set a *frequency threshold* to determine the qualification of a lane reversal for intensification—if the residence frequency of a full lane reversal has been greater than the predefined threshold since the first time it appears in the search trajectory, its existence should be fixed in the subsequent solutions until a diversification move is conducted. In other words, we do not need to consider a possible change of this “locked” roadway subnetwork any longer in the succeeding neighborhood searches. This specific intensification instance belongs to the *intensification by decomposition* strategy, named by Glover and Laguna (1997).

Of course, the residence frequency is not meaningful, if the denominator, that is, the total number of iterations used to calculate the residence frequency, is too small. A minimum denominator value needs to be set a priori for evaluating the validity of a residence frequency value. That is, the residence frequency of a solution element (i.e., a lane reversal configuration of some roadway subnetwork) begins to be counted only after the number of iterations exceeds a minimum number.

The advantage of applying such a simple intensification measure is to reduce the size of a neighborhood region and let the subsequent search concentrate on the remaining search space more thoroughly. In our experiments, it is found that many links close to network egress nodes or on some primary evacuation routes quickly obtain the intensification qualification for the full lane-reversal assignments with their outbound directions. As for partial lane reversals, however, there is no such a reliable intensification measure identified in our study.

A high residence frequency with a specific lane reversal in some roadway subnetwork may indicate that this lane-reversal configuration is highly attractive, or may indicate the opposite result, if its associated iterations correspond to low-quality solutions. On the other hand, a high residence frequency that is high when there are both high- and low-quality solutions may point to an entrenched attribute that causes the search space to be restricted, and that needs to be jettisoned to allow increased diversity ([Glover and Laguna, 1997](#)). To judge the necessity of diversifications, we need to jointly investigate both the residence frequency and the solution quality associated with those lane reversals implemented along the search itinerary. The motivation for applying diversification for our lane-reversal optimization subproblem is, when a large number

of iterations have been conducted without any improvement to the objective function value, it may be more attractive to transfer our search into a distant unexplored region than to continue the current local search.

To encourage diversification, we implement the following diversification means in our search procedure: once the number of iterations that do not improve the objective function value exceeds a pre-specified value, we turn to examine all the full lane reversals that have been confirmed by the previous intensification operations with their intensification history and associated solution qualities; the set of full lane reversals that produce a large number of relatively low-quality solutions are reversed fully as a diversification move. On some degree, this diversification change may be regarded as a reverse manipulation to the intensification operation. We set the intensification to always yield the diversification, that is, when a diversification move arises, the intensification status of all intensified elements (i.e., those “locked” roadway subnetworks) are released. Under this setting, intensification is used for local neighborhood searches while diversification for starting a new search in an unexplored region.

4.3.5 Lagrangian multiplier updating

The Lagrangian multiplier updating mechanism is critical to the feasibility and optimality of the solutions derived by the LR-TS heuristic procedure. A too low value of a Lagrangian multiplier in the penalty term may result in a final infeasible solution; a too high value may lead to the search process to deviate away from the optimal solution (in spite of other suboptimal conditions caused by the heuristic procedure). A simple but effective iteration-based self-adjusting method for the multiplier updating

is set based on the use of another residence frequency that records the number of each intersection crossing point existing in solutions during the search process. If a crossing point consecutively exists in the solution itinerary (e.g., 5 times), its corresponding unit penalty cost (i.e., its Lagrangian multiplier) is increased; otherwise, the penalty cost is decreased. A unit updating cost is accordingly set up to specify the increment/decrement amount each time, whose value is dependent on the particular target problem.

4.3.6 Stopping rule

Tabu search is by nature an open-ended search strategy without a convergence or optimality criterion. Theoretically, a tabu search procedure could go on forever unless the optimal solution of the target problem is known in advance. Thus, some exogenous stopping criteria are needed to terminate a tabu search process, such as a pre-specified number of iterations, a pre-specified number of iterations without an improvement in the objective function value, or a pre-specified objective threshold value is reached. In our case, we set the search termination criterion by using a pre-specified number of diversification phases and in each diversification phase using a pre-specified number of non-improving iterations to stop a current local search and start a distant diversification search.

4.4 Estimation of the network flow pattern

As we pointed out in the beginning of this chapter, an evaluation of the objective function of the Lagrangian problem in the tabu search process turns to two subsequent computational procedures: 1) first, a stochastic traffic assignment in the given reduced

evacuation network, which has been defined as the lower-level problem in the proposed bi-level network optimization model (see (3.11)-(3.22)); 2) second, an independent traffic crossing-elimination examination for each intersection of interest, given the network flow pattern derived from the preceding traffic assignment process.

This section discusses the issues of solving the traffic assignment subproblem. In the literature, the stochastic user-equilibrium traffic assignment has been carried out by two types of network loading methods: logit-based and probit-based methods. The difference between the two methods is due to the statistical assumption of the random travel cost component: a logit model is based on the Gumbel distribution and a probit model uses the normal distribution. The probit-based loading method is preferable to the logit-based because it can properly take into account the overlapping or correlated network proportions when estimating the route cost distributions and route choice probabilities. However, there exists no closed form of exact methods for computing the route choice probability in a probit-based network loading. Previous research suggested two strategies of implementing the probit-based network loading: Monte Carlo simulation (Sheffi and Powell, 1982; Sheffi, 1985) and Clark's approximation (Maher, 1992; Maher and Hughes, 1997).

In view of the given behavioral implication and computational cost, we suggest using the analytical probit-based traffic assignment algorithm introduced by Maher and Hughes (1997) for the stochastic traffic flow pattern estimation, in which the network loading procedure is powered by Clark's approximation. This method has an analogous algorithmic procedure to the convex combinations method that was applied to solve the deterministic user-equilibrium traffic assignment problem in 1970s (refer to LeBlanc et al., 1975).

The algorithmic procedure of this analytical traffic assignment method can be briefly depicted as follows:

Step 0. Choose a set of initial arc travel times $\mathbf{t}^{(0)}$, usually free-flow travel times. Find an initial traffic flow pattern $\mathbf{x}^{(1)}$, by performing a stochastic network loading based on the initial arc travel times $\mathbf{t}^{(0)}$. Set $k = 1$.

Step 1. Calculate the updated arc travel times $\mathbf{t}^{(k)}$ with the given traffic flow pattern $\mathbf{x}^{(k)}$ using the arc performance function, i.e., $\mathbf{t}^{(k)} = \mathbf{t}(\mathbf{x}^{(k)})$.

Step 2. Find the auxiliary traffic flow pattern $\mathbf{x}'^{(k)}$, by performing a stochastic network loading based on the current arc travel times $\mathbf{t}^{(k)}$.

Step 3. Make a line search to find the optimal value of step length λ_k so as to minimize $w(\mathbf{x})$ along the search direction $\mathbf{x}'^{(k)} - \mathbf{x}^{(k)}$, where $w(\mathbf{x})$ is the objective function of the equivalent program to the stochastic user-equilibrium assignment,

$$w(\mathbf{x}) = \sum_{\eta_l} x_{\eta_l} t_{\eta_l}(x_{\eta_l}, n_{\eta_l}) - \sum_{\eta_l} \int_0^{x_{\eta_l}} t_{\eta_l}(\omega) d\omega - \sum_{rs} q_{rs} E\left[\min_k(T_k^{rs}) \mid \mathbf{t}^{rs}(\mathbf{x})\right]. \quad (4.6)$$

Step 4. Calculate the updated traffic flow pattern: $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k(\mathbf{x}'^{(k)} - \mathbf{x}^{(k)})$.

Step 5. If a convergence criterion is met, stop the iteration and conclude that the current traffic flow pattern $\mathbf{x}^{(k+1)}$ is the set of stochastic user-equilibrium flows; otherwise, set $k = k + 1$ and go to step 1.

The stochastic magnitude of individuals' travel time perceptions is specified by a travel time variability parameter (i.e., the ratio of the standard deviation to the mean of the travel time), which can be interpreted as a proportionality indicator of the travel time variance to the mean. Many previous travel time studies showed that this travel

time variability parameter is a variable related to the traffic congestion conditions, where its value is relatively lower when traffic is very heavy or light than moderate. In an evacuation network, it is expected that the traffic is quite congested. For the sake of simplicity, we arbitrarily set the value of this travel time variability parameter equal to a modest value, 0.5. Moreover, this constant variability parameter is also universally applied to all the networks used in this study.

The primary computational component in the above procedure is the traffic network loading, which needs to be performed at each iteration. Some implementation issues of this computational component are discussed in [Appendix A](#).

4.5 Examination of the intersection crossing elimination

The traffic flow pattern resulting from the above traffic assignment process in a reduced network does not give all the required information about the turning movements at intersections. In fact, the representation of an intersection as a node ignores the crossing-elimination configuration subproblem and regards the intersection as a “black box”. To completely evaluate the objective function of the Lagrangian problem, we also need to compute the value of the penalty term, which is equal to checking the crossing-elimination violation conditions at all the intersections of interest subject to a given traffic flow pattern in the reduced network. For this purpose, we formulated a linear mixed integer programming model for the crossing-elimination configuration problem. The objective of this program is to minimize the number of crossing points between turning movements with positive flow rates at the intersection; the constraints include the integral constraints, capacity constraints, and flow reservation constraints that are transplanted from the original problem. For a typical

four-leg intersection such as the one shown in [Figure 3.1](#), the functional form of this integer optimization problem is,

$$\min \quad z(\mathbf{y}) = \sum_{\eta, \rho\sigma} (y_{\eta\iota} + y_{\rho\sigma} - 1)^+ \quad (4.7)$$

$$\text{s.t.} \quad y_{\eta\iota} = 0 \text{ or } 1, \quad \forall \eta \rightarrow \iota \quad (4.8)$$

$$x_{\eta\iota} \leq u_{\eta\iota} y_{\eta\iota}, \quad \forall \eta \rightarrow \iota \quad (4.9)$$

$$x_{\eta\iota} \geq 0, \quad \forall \eta \rightarrow \iota \quad (4.10)$$

$$x_{\iota\mathcal{S}} - \sum_{\eta \in S_\iota} x_{\eta\iota} = 0, \text{ and} \quad \forall \iota \rightarrow \mathcal{S} \quad (4.11a)$$

$$x_{\tau\rho} - \sum_{\sigma \in R_\rho} x_{\rho\sigma} = 0, \quad \forall \tau \rightarrow \rho \quad (4.11b)$$

where S_ι and R_ρ represent the set containing the starting nodes of all the arcs pointing to node ι and the set containing the ending nodes of all the arcs emanating from node ρ , respectively, i.e., $S_\iota = \{\eta, \varphi, \mu\}$ and $R_m = \{\sigma, \nu, \gamma\}$, and the flow rates $x_{\iota\mathcal{S}}$ and $x_{\tau\rho}$ are the input of this program, which are given by the current lane-reversal solution.

The objective function of this local optimization program serves as a surrogate of the crossing-elimination constraints and counts the number of crossing points if any of the constraints is violated. This conversion can be seen as, for example, $y_{\varphi\iota} + y_{\mu\gamma} \leq 1$ is relaxed and $(y_{\varphi\iota} + y_{\mu\gamma} - 1)^+$ is inserted into the objective function; or $y_{\eta\iota} + y_{\rho\sigma} + y_{\mu\gamma} \leq 1$ is relaxed and instead, $(y_{\eta\iota} + y_{\rho\sigma} - 1)^+$, $(y_{\rho\sigma} + y_{\mu\gamma} - 1)^+$ and $(y_{\mu\gamma} + y_{\eta\iota} - 1)^+$ are added. The optimal objective function value of this program indicates the feasibility of a resulting intersection flow pattern: if the value is zero, the optimized intersection flow pattern subject to the current lane-reversal solution is a feasible solution;

otherwise, there is one or more crossing points at this intersection and a penalty value should be imposed to the objective function of the Lagrangian problem.

The crossing-elimination subproblem can be efficiently solved using the branch-and-bound method due to its relatively small search space. In the case of a typical four-leg intersection, it has only 8 binary integer variables and 8 real variables with 8 capacity constraints and 8 flow conservation constraints. This method follows a vertex-and-branch tree structure, where the linear relaxation subproblem at each vertex is used to establish the lower bound for the feasible region corresponding to the vertex. Two simple algorithmic choices can be applied to accelerate the branch-and-bound search for this linear mixed integer program. To see these, once again, let us refer to [Figure 3.1](#). We can observe that, for example, first, if $x_{\kappa} = 0$, we immediately have $x_{\eta_l} = 0$ and $y_{\eta_l} = 0, \forall \eta_l \in A_l$; second, if $x_{\kappa} > 0$, assign as much flow to x_{μ_l} as possible, where x_{μ_l} denotes the flow amount on the right-turn arc $\mu \rightarrow l$ arriving at node l , since a right turn would not cause any crossing conflict. Application of these rules at the beginning of a branch-and-bound search can effectively reduce the remaining search space.

We also developed a more efficient simplex-based iterative solution method to solve this crossing-elimination subproblem. The rationale and proof of this method are elaborated in [Appendix B](#). Its algorithmic procedure is sketched as follows:

Step 1. Obtain a starting basic feasible solution as the current solution and compute its objective function value z^* . In view of the problem structure similar to that of the classic transportation problem, this can be conveniently accomplished by applying the northwest corner rule in the tableau (see Bazaraa et al., [1990](#)).

Step 2. Conduct all the candidate pivot moves by entering each nonbasic variable into the basis and compute the updated objective function value with each candidate move. Choose the best move with the lowest objective function value z' .

Step 3. Compare the objective function value with the best move, z' , and the current objective function value, z^* . If $z' \geq z^*$, stop the iteration and we have the optimal solution z^* at hand; if $z' < z^*$, implement the best move to obtain the updated basic feasible solution and assign $z^* = z'$, and then go to step 2.

Solving the crossing-elimination optimization subproblem is indeed a local network design problem and a traffic re-assignment process for the intersection subnetwork. Such a subnetwork change is certainly a change to the expanded network. This change, however, will not cause a change of the traffic flow pattern obtained from the link-based stochastic traffic assignment in the reduced network. In other words, the traffic flow pattern obtained from the reduced network can still be maintained in the expanded network with the intersection crossing reduction/optimization. This conclusion holds subject to a *homogeneous* flow requirement that is supported by two specific modeling settings in our problem. This requirement is a sufficient (but perhaps not necessary) condition to the conclusion.

The first setting is that the underlying stochastic traffic assignment algorithm used for generating the traffic flow pattern implies the Markovian routing behavior that any individual would choose his remaining route to the destination without considering the route he has experienced between the origin and his current location. The resulting traffic flow pattern possesses the property that the traffic flow arriving at any intermediate node in a network is assigned as if this node is a destination. As we described previously, an analytical network loading algorithm based on Clark's

approximation (Maher, 1992; Maher and Hughes, 1997) is employed to approximate the probit-based stochastic user-equilibrium traffic flow pattern[†]. The underlying individual route choice behavior within this approximation procedure does possess the Markovian property, which virtually assures the traffic flow merging at any intermediate node is homogeneous by origin.

The second setting is the one-destination network representation. An immediate result from this setting is that all individuals departing from or arriving at any single source or intermediate node in the network go to the same destination. From a modeling perspective, this result guarantees that all individuals going through a node are in a homogeneous population with a single route choice function (that implies an identical route choice probability distribution with each individual). Note that in a general multi-commodity network (i.e., a network with multiple origins and destinations), the crossing-elimination optimization process for an intersection may change the paths of traffic flows going through the intersection, and so the destinations of these path flows. The occurrence of a destination change would possibly result in an infeasible traffic flow pattern[‡]. However, this phenomenon will not occur in a network with the one-destination setting, or, in other words, the traffic flow diverging at any intermediate node is homogeneous by destination.

As a result, from the two settings, we can conclude that the traffic flow between any intermediate node and the destination mode (in the reduced network) can be regarded as a homogeneous flow pattern as if it is assigned between these two nodes. As long

[†] In contrast, a traffic assignment implying the whole-path routing behavior, for example, the stochastic user-equilibrium assignment using path enumeration, does not hold this conclusion.

[‡] By *infeasible*, we mean that the resulting traffic flow pattern caused by the crossing-elimination optimization process may not satisfy the flow conservation constraints of the original problem.

as the (arc-based) traffic flow pattern holds, any individual's Markovian route choice behavior would not be changed.

We highlight the above conclusion in the following. Given the implied Markovian routing behavior assumption and single-destination network setting, the connectivity change in an intersection subnetwork subject to the constraints (4.8)-(4.11) does not change a traffic flow pattern assigned in the reduced network. This is the underlying reason that we are able to optimize the network design at each individual intersection alone. Given this conclusion, the stochastic traffic assignment process for evaluating the objective function can be always conducted based on the reduced network.

With synthesizing the pieces of knowledge obtained above, we know that for each evaluation of the objective function, computing the two terms of the objective function, which is equivalent to two separate optimization subproblems, can be conducted in a sequential manner, in which the crossing-elimination subproblem is subject to the result of the corresponding lane-reversal subproblem.

4.6 The algorithmic procedure

As an overview of the integrated Lagrangian relaxation and tabu search procedure, we compile all the algorithmic elements into the following pseudo-code form. For the sake of concision, only major algorithmic steps are presented. Details of many subroutines are simply condensed as single clauses and can be referred to in the above text.

```

algorithm LR-TS heuristic;

begin

    define elite_size, tabu_tenure, freq_threshold, max_iteration_number,
        max_diversification_number;

    define roadway subnetwork set  $R = \{r\}$ , intersection subnetwork set  $T = \{t\}$ ,
        crossing-elimination constraint  $C(t) = \{c\}$  for each  $t \in T$ ;

    initialize  $i := 0$ , best_solution, unit_penalty( $c$ ): = 0 for each  $c \in C(t)$ ,  $t \in T$ ;

    while  $i < \text{max\_diversification\_number}$  do

        begin

            create tabu_list, residence_freq;

             $j := 0$ ;

            while  $j < \text{max\_iteration\_number}$  do

                begin

                    create elite_list;

                    for each  $r \in R$  do

                        begin

                            if residence_freq( $r$ ) < freq_threshold

                                begin

                                    identify a candidate move, move( $r$ );

                                    evaluate the objective function of move( $r$ ), obj(move( $r$ ));

                                    if move( $r$ ) belongs to tabu_list then

                                        begin

                                            cancel the candidacy of move( $r$ );

                                            if obj(move( $r$ )) < obj(move(best_solution)) then

```

Figure 4.3 The algorithmic procedure of the LR-TS heuristic

Figure 4.3 (Continued)

```

    begin
        retrieve the candidacy of move(r);
    end;
end;
end;
update elite_list;
end;
end;
for each move(e) ∈ elite_list do
begin
    conduct a local move, move(e), and evaluate obj(move(e));
    update unit_penalty(c) for each  $c \in C(t)$ ,  $t \in T$ ;
    update tabu_list;
    update residence_freq;
    update best_solution;
     $k = k + 1$ ;
    if  $obj(move(e)) \geq obj(best\_solution)$  then  $j = j + 1$  else  $j = 0$ ;
end;
end;
conduct a diversification move;
 $i = i + 1$ ;
end;
end;
```

4.7 A numerical example

A simple numerical example represented by its reduced network topology is presented in [Figure 4.4](#). We use this example to demonstrate the effectiveness of the proposed solution strategy, in which Lagrangian relaxation provides the decomposition mechanism to separate the lane-reversal and crossing-elimination subproblems and tabu search serves as the algorithmic search engine. This example network shows a ∞ -shape topology, where the original configuration of this network has two-way connections on all this roadway sections. All the source nodes, i.e., nodes 5-14, are located intermediately on roadway arcs, while there is a single destination node, i.e., node 4, representing the location of a network exit or a shelter. For the sake of simplicity, it is assumed that all the directed arcs have only one single lane. Therefore, any lane-reversal operation in this network will result in both network capacity and connectivity changes. Specifically, node 2 represents a typical four-leg intersection with “U”-turn prohibited. The intersection subnetwork at node 2 has the same network structure as the one we showed early in [Figure 3.1](#). As we will see below, traffic crossing conflicts may occur at this intersection for some solutions on the search itinerary. The network information is given in [Figure 4.4\(a\)](#) and the demand data are labeled beside their respective origin nodes in [Figure 4.4\(b\)](#).

In applying our heuristic search procedure, we arbitrarily set all the unit penalty costs (or Lagrangian multipliers) equal to 2×10^4 time units. In accordance with the previous discussion, the probit-based stochastic routing behavior is assumed to underlie any evacuation flow pattern in this network, where the stochastic magnitude of an evacuee’s travel time perception is denoted by a proportional constant that is defined as the standard-deviation-over-mean ratio. This proportional constant is set as

0.5. Other algorithmic parameters used in this heuristic procedure include tabu tenure = 4 and the allowed number of non-improving iterations = 20. No intensification and diversification strategies are used in view of the overly simple structure and relatively small size of the problem. Also, the BPR function is used to calculate link travel times.

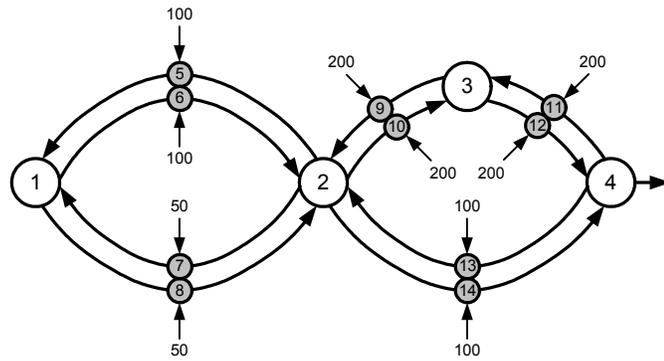
A number of first iterations generated by the integrated search procedure starting from the original network configuration are illustrated in [Figure 4.4](#). These evacuation network solutions to the Lagrangian problem are represented in the reduced networks that only show the lane-reversal configurations. The crossing-conflict violation condition with each network solution can be easily assessed by using the minimum number of crossing-conflicts as well as the traffic flow rates with these crossing traffic movements. In [Figure 4.5](#), we show the optimal traffic movement configuration in the only intersection subnetwork (i.e., node 2 in the reduced network), as subject to the corresponding lane-reversal configuration as well as the traffic flow pattern at each iteration. The network performance of each solution is indicated by the objective function value of the Lagrangian problem, which is equal to *total travel time + total penalty cost*.

Link pair	Free-flow travel time	Capacity	Number of lanes
1-6-2	5	100	1
2-5-1	5	100	1
1-8-2	5	100	1
2-7-1	5	100	1
2-10-3	5	200	1
3-9-2	5	200	1
3-12-4	5	150	1
4-11-3	5	150	1
2-14-4	10	200	1
4-13-2	10	200	1

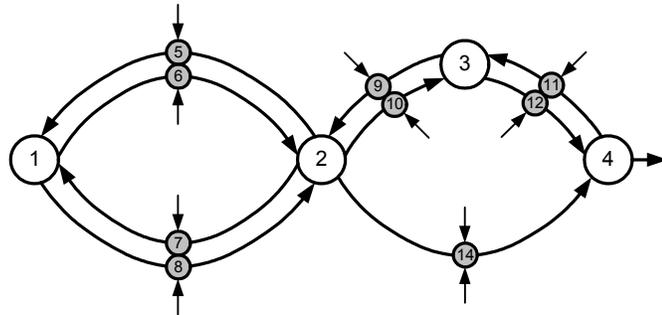
(a) Network information of the illustrative numerical example

Figure 4.4 Network information and iterative solutions of the numerical example

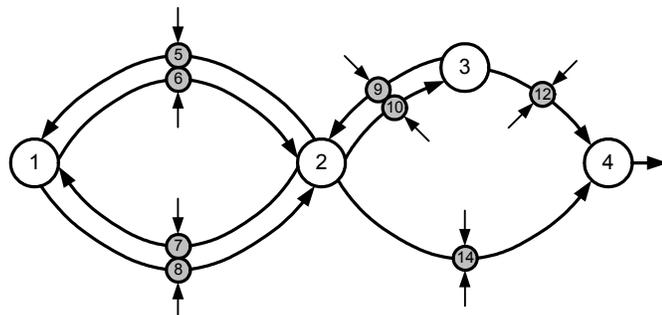
Figure 4.4 (Continued)



(b) Iteration 0 (The objective function value: $3.82 \times 10^5 + 2 \times 10^4$)

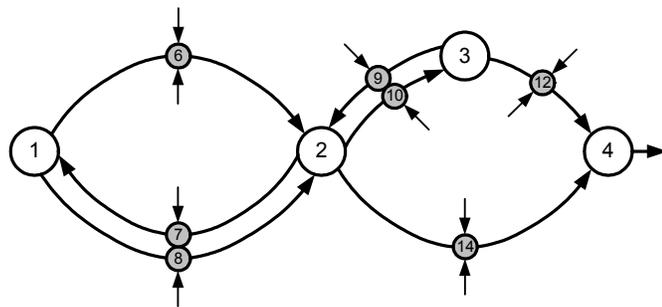


(c) Iteration 1 (The objective function value: $1.01 \times 10^5 + 4 \times 10^4$)

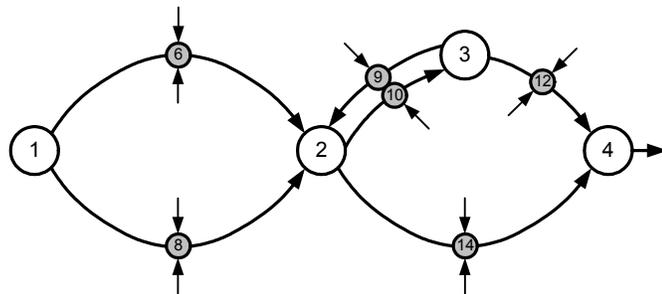


(d) Iteration 2 (The objective function value: $4.51 \times 10^4 + 2 \times 10^4$)

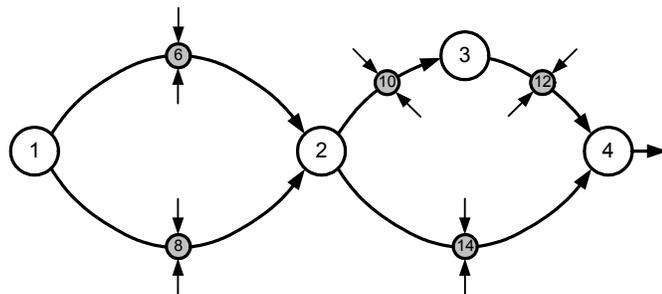
Figure 4.4 (Continued)



(e) Iteration 3 (The objective function value: $4.31 \times 10^4 + 2 \times 10^4$)



(f) Iteration 4 (The objective function value: $4.24 \times 10^4 + 2 \times 10^4$)



(g) Iteration 5 (The objective function value: $5.12 \times 10^4 + 0$)

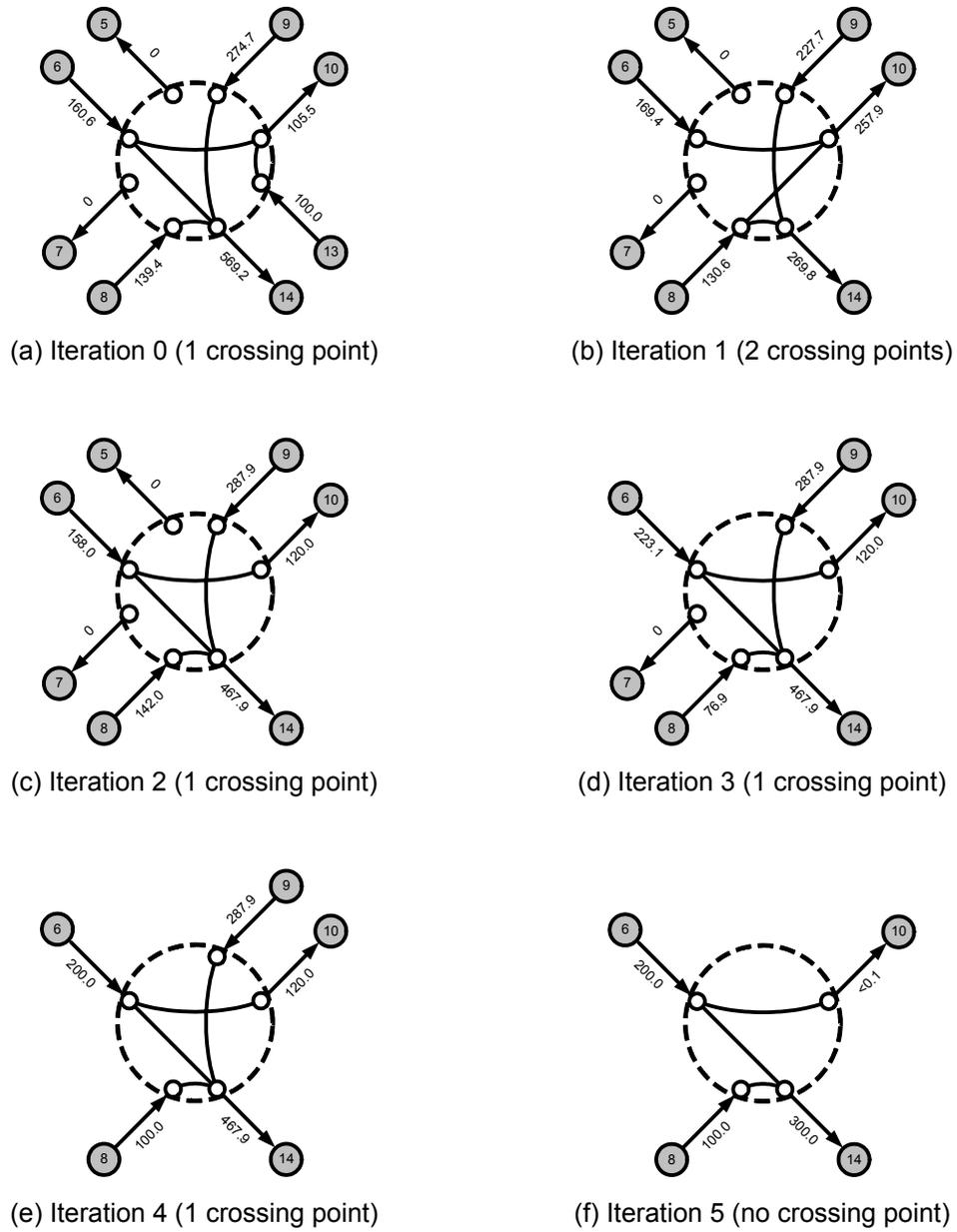


Figure 4.5 Iterations of the optimal traffic movement configuration in the example intersection subnetwork

For the purpose of illustration, we only presented here a sequence of first solutions with descending objective function values, including the initial network configuration

and its subsequent five iterations. It can be observed that the search procedure in turn reverses arc pairs $4 \rightarrow 13 \rightarrow 2$, $4 \rightarrow 11 \rightarrow 3$, $2 \rightarrow 5 \rightarrow 1$, $2 \rightarrow 7 \rightarrow 1$ and $3 \rightarrow 9 \rightarrow 2$, and identifies at iteration 5 the best solution to the Lagrangian problem with the predefined unit penalty cost. In fact, for this simple problem, we can identify by enumeration without an intensive computational effort that this solution is the optimal solution to the Lagrangian problem and also the optimal solution to the original network optimization problem. The total travel time corresponding to this solution is 5.12×10^4 time units and all the crossing-elimination constraints are satisfied under this network configuration (refer to [Figure 4.5\(f\)](#)).

The success of the integrated heuristic search method is greatly related to the choice of the unit penalty cost value, P . In our case, an arbitrary choice of setting P equal to 2×10^4 time units results in that the first local optimum encountered in the search itinerary is actually the global optimal solution. Taking a closer look at the iterations will reveal the following phenomenon: if we preset $P < 0.88 \times 10^4$, the optimal solution we find for the Lagrangian problem is the one obtained at iteration 4, which is actually an infeasible solution to the original problem; if we preset $P > 2.81 \times 10^5$, we would not choose the solution shown at iteration 2 as the next best solution after iteration 1 and accordingly go on with the search along a different, possibly less efficient path and it might prevent us from visiting the true optimal solution at all. Clearly, an appropriate range for a fixed unit penalty cost value to achieve the desirable search itinerary in this example is $0.88 \times 10^4 \leq P \leq 2.81 \times 10^5$.

By examining the search iterations of the illustrative problem, it is clear that the proposed search strategy can effectively reduce the combinatorial complexity imposed by the integration of the lane-reversal and crossing-elimination constraints while

successfully avoid the possible infeasibility trap introduced by the relaxation through setting an appropriate penalty cost value. Although the example problem is successfully and readily tackled by the heuristic method, many features and advantages of our algorithmic design, however, may not be embodied by the solution search process applied to this overly simple example. In fact, a simple greedy descent heuristic without resorting to sophisticated algorithmic designs can also be used to solve this example problem optimally. To gain a comprehensive evaluation on the efficiency and effectiveness of the proposed Lagrangian-based, tabu-powered heuristic method, we need to carry out experiments in large networks with realistic topology. In the next chapter, we present some preliminary computational results from implementing our heuristic algorithm in a number of such larger networks.

CHAPTER 5

ALGORITHM CALIBRATION AND EVALUATION

*A witty statesman said, you might prove anything by figures.
—Thomas Carlyle*

The performance of any metaheuristic is highly dependent on the proper calibration of its algorithmic parameters. Our search heuristic is not an exceptional case. A calibration phase is required prior to the implementation of the developed algorithmic procedure in large-scale problems and thus it becomes an integral part of the development of the solution procedure.

As we described in the previous chapter, the solution procedure is an integrated process of solving the relaxed Lagrangian problem and updating the Lagrangian multipliers. Although the Lagrangian multiplier updating mechanism requires a specification of the increment/decrement value for the unit penalty cost, this value is a problem-specific parameter and is relatively less sensitive to the algorithm performance[†]. A proper initial unit penalty cost and an increment/decrement cost value for a given problem instance can be readily obtained by trying a limited number of preset candidate values.

In this chapter, therefore, our focus is to evaluate and calibrate a set of parameters used to tailor the tabu search procedure for the Lagrangian relaxation problem. In terms of the applicable levels of these parameters, this set includes two parts: 1) local search parameters; 2) diversification search parameters. We expected to determine a

[†] In many cases, a fixed unit penalty cost, or in other words, an increasing/decreasing cost value equal to 0, works perfectly for the problem solutions.

common set of parameter values that can lead to the optimal or near-optimal solutions for the type of network optimization problems of interest here.

The purposes of this chapter are threefold: 1) to calibrate the algorithmic parameters and characterize the search behavior of the heuristic procedure; 2) to gain insights about the efficiency and effectiveness of the heuristic algorithm for a set of problem instances with a variety of demand and supply settings; and 3) to establish the fidelity of the heuristic algorithm in solving the problem type of interest here.

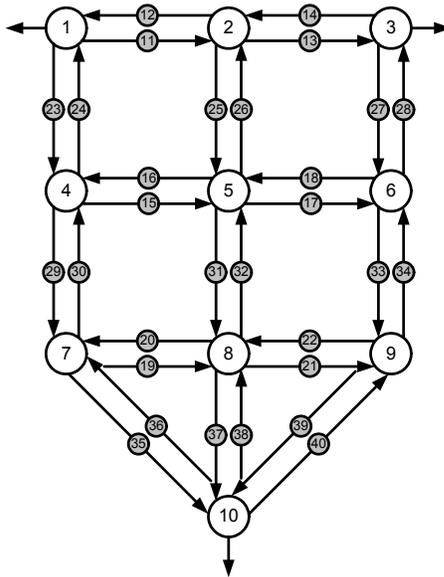
5.1 Experimental problem instances

The parameter calibration is essentially a multi-objective, multi-dimensional optimization process, in which we expect to identify a proper set of algorithmic parameter values so as to minimize the gap between the optimal solution and the best solution achieved by the heuristic procedure. Because of the combinatorial effect in choosing among the discrete values of the parameters, this process requires a relatively large number of repeated performance evaluations, where each evaluation resorts to an application of the whole LR-TS search process for the problem instance with a set of pre-specified candidate parameter values and it often involves some empirical judgment on the parameter value comparison and selection. Given the limited computational resources, it is not possible to include a large number of problem instances for the calibration. Instead, it is desirable to limit the number and size of problem instances used for the calibration but manage them to cover the diverse problem settings we expect to encounter in realistic evacuation networks.

A set of problem instances of relatively small size but with different supply and demand characteristics (see [Figure 5.1](#)) are selected for the parameter calibration. The travel demand amount and distribution information of these networks is also included.

The first three networks are synthetic evacuation networks. We arbitrarily set all the links in these networks with two lanes and each lane with the capacity of 200 vehicles per time unit. The size of these synthetic examples ranges from 40 nodes and 60 links to 85 nodes and 128 links (in their reduced versions). The number of egress nodes in these networks range from 3 to 12, representing different egress capacities.

The fourth network is a surface traffic network located in Sioux Falls, South Dakota, which in the literature has been used in a number of network design studies (for example, see [LeBlanc \(1975\)](#)). This network comprises the major arterial roadways of Sioux Falls, which shows a typical grid structure of urban networks. In its original version, the network consists of 24 nodes and 76 links. Due to the setting of the intersection crossing-elimination control, travel demands need to be reassigned from the intersection nodes to the intermediate nodes along links. As part of our modeling work in formulating a lane-based network optimization problem, such a network supply and demand modification is made that a source node is added to each eligible link and traffic demand is accordingly redistributed to these newly added source nodes from the original source nodes that actually represent intersections or interchanges. The modified topology of the Sioux Falls network, as shown in [Figure 5.1\(d\)](#), is of 100 nodes and 152 links.

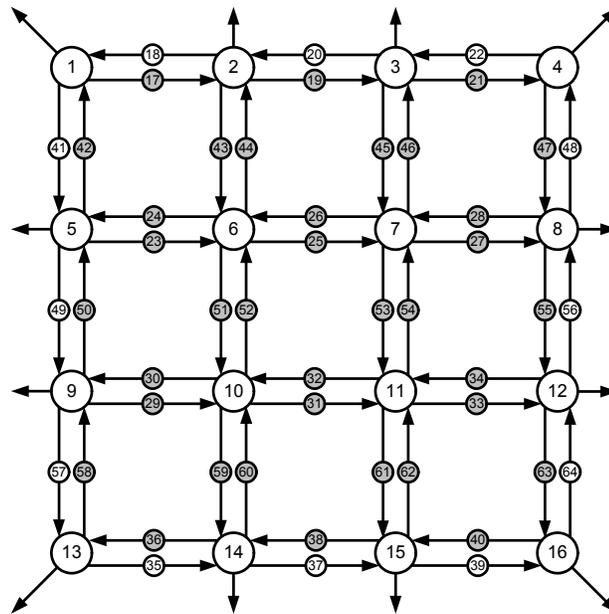


Node	Demand	Node	Demand	Node	Demand
11	266	21	467.2	31	439.8
12	406.5	22	506.1	32	353.8
13	514.1	23	374.8	33	377.4
14	318.5	24	462.2	34	321.8
15	291.6	25	536.8	35	429.9
16	386.6	26	299.1	36	280
17	485	27	250.7	37	480.6
18	330.8	28	441.6	38	307
19	284	29	284.7	39	541.7
20	367.2	30	429.4	40	377.5

(a) Problem instance 1: A synthetic small network (40 nodes and 60 links)

Figure 5.1 The problem instances used for the algorithmic parameter calibration

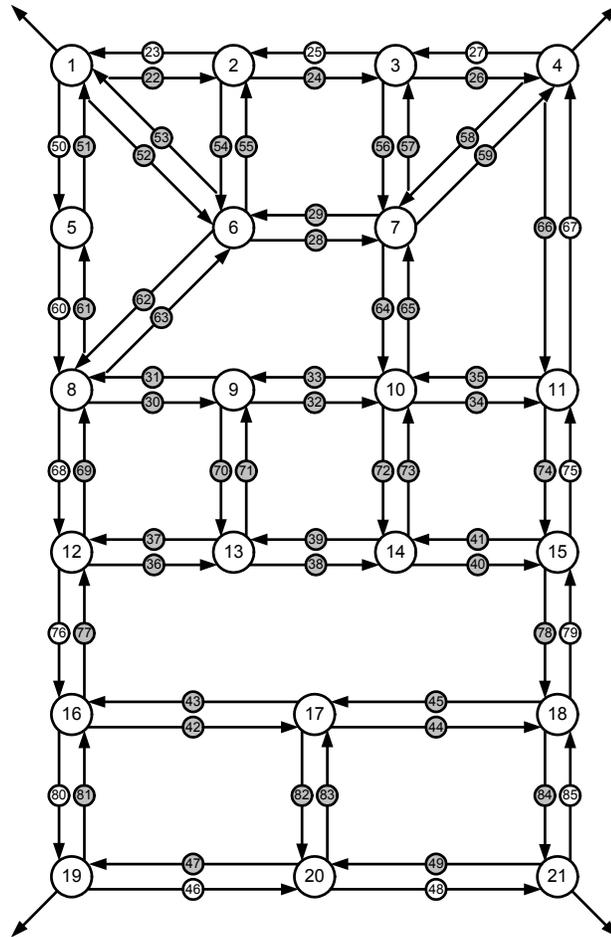
Figure 5.1 (Continued)



Node	Demand	Node	Demand	Node	Demand
17	314.6	23	446.9	29	392.1
24	279.9	30	309.6	36	331.8
42	388	50	403.6	58	297.5
43	393.5	51	337.2	59	276.6
19	448.5	25	453.5	31	428.9
26	340.5	32	455.3	38	357.1
44	385	52	379.6	60	405.6
45	356.5	53	194.6	61	363.2
21	392.1	27	413.3	33	353.5
28	553.5	34	559.3	40	437.1
46	339.4	54	501.8	62	472.8
47	265.3	55	242	63	611.2

(b) Problem instance 2: A synthetic grid network (64 nodes and 96 links)

Figure 5.1 (Continued)



(c) Problem instance 3: A synthetic urban network (85 nodes and 128 links)

Figure 5.1 (Continued)

Node	Demand	Node	Demand	Node	Demand
22	184.2	63	285.3	74	293.9
53	218.8	64	261.6	36	149.1
54	236.6	59	156.9	38	273.8
24	242.9	35	193	40	251.1
29	281.2	65	241	43	230.7
55	207.1	66	231.1	45	160.3
56	258.6	30	105.1	77	118.5
26	191	37	283.2	78	269.4
57	147.7	69	281.2	42	134.4
58	276.9	70	153.1	47	102.8
51	143.4	32	188.2	81	252.7
52	249	39	180.5	82	130.1
61	258.7	71	265.5	44	179.6
62	240.4	72	164.7	49	234.9
28	235.9	34	258.4	83	213.9
31	280.8	41	245.7	84	148.9
33	216	73	228.1		

(c) Problem instance 3: A synthetic urban network (85 nodes and 128 links)

Figure 5.1 (Continued)

Node	Demand	Node	Demand	Node	Demand
25	66.7	71	222.9	93	184.4
28	135.4	35	150	81	239.6
30	184.4	44	385.9	48	178.1
32	103.1	72	222.9	80	229.2
58	60.4	73	313	85	188.5
59	60.4	37	188.6	41	253.7
27	75	52	116.1	82	239.6
38	188.6	76	148.4	83	257.8
62	87	77	263	47	178.1
63	176.6	87	297.9	96	175.3
29	92.2	91	191.2	90	238.6
40	351.6	39	351.6	97	163
64	176.6	46	184.9	95	175.3
65	200.5	78	189.6	56	105.5
69	404.2	79	346.9	94	184.4
31	71.4	45	184.9	43	385.9
34	171.4	50	202.6	84	379.7
66	116.2	88	149	86	377.1
67	126.6	89	238.6	98	229.7
33	171.4	49	202.6	99	298.2
42	371.4	54	97.4		
70	319.8	92	115.6		

(d) Problem instance 4: The Sioux Falls network (100 nodes and 152 links)

We also supposed in this network such an evacuation event that, for example, a harmful chemical or radioactive source is found at a central place of the city. An emergency evacuation plan needs to be devised and implemented so as to evacuate all the residents from the emergency area. No shelter or refuge for accommodating evacuees is designated in advance. The emergency management agency would order everyone to leave the network through three major exits connecting the highways to the north of the network. As consistent with this evacuation order, the integrated destination and route choice concept is accordingly assumed in this case, by which a

super dummy destination node is added to the network and all the three egress nodes are virtually connected to this dummy node.

It is our expectation that the whole set of example networks can, to some extent, imitate a number of different urban traffic network types. These network settings include network topologies, roadway capacities, demand levels, and egress capacities and distributions, while avoiding an excessive calibration effort. The selection of these networks of relatively small size can make us readily track the search behavior of the algorithmic procedure and analyze the sensitivity of the calibrated parameters to the search performance.

5.2 Local search calibration

In our tabu search method, the key algorithmic parameters to be calibrated for local searches include elite capacity (*elite_capacity*), tabu tenure (*tabu_tenure*) and residence frequency threshold (*freq_threshold*). These parameters are important to the performance of our heuristic search procedure, on either the solution quality or the search efficiency, or both of them. A conservatively small elite capacity may generate a sequence of iterations that implement the best move each time in terms of the results of an exhaustive network scan process. However, such a setting may not be able to sufficiently exploit the information implied by the network scanning results obtained each time and to efficiently conduct moves in the following step. An overly large elite capacity, on the other hand, often provides biased information in choosing eligible roadway links for subsequent lane reversal operations. Tabu tenure, indicating the size of a tabu list, is another key parameter that greatly influences the performance of the tabu search procedure. An excessively small tenure value may not be able to

effectively prevent local cycling occurrences, while a too large tenure value may prevent the search from entering some potential promising regions. As for residence frequency threshold, it signifies the intensification sensitivity. A too low frequency threshold would limit the search in an overly small region and hence lead it to be trapped at some local optimal point. A too high frequency threshold might be useless in performing the intensification function and achieving the benefit of accelerating the search.

We expect to determine a robust set of values or value ranges for these parameters so that the calibrated heuristic procedure can perform well over a broad range of evacuation network optimization problems of the type defined in this research. The aim is to develop a widely accepted parameter criterion set that maintains a good counterbalance between the solution quality and search efficiency and provides a consistent performance level over the problems tested here.

The solution quality is directly evaluated with the objective function value of a solution. While the search efficiency can be generally surrogated by the computing time, this efficiency performance measure, however, is highly dependent on the performance of the used computing facility and coding platform. A more universal way to evaluate the computational effort in many combinatorial optimization problems is to count the number of times of evaluating the objective function during the search itinerary if the evaluation itself dominates the computational cost during the whole search process. This criterion is applicable to our case. In fact, in our network optimization problem, the objective function evaluation that is essentially a computational process for solving a network-wide traffic assignment problem and a set of intersection-wide traffic movement optimization problem is the computational

bottleneck in the search procedure. In contrast, other algorithmic operations on the local search level, such as lane exchange, demand exchange, elite list and tabu list updating, and so on, or on the diversification search level, such as diversification move, only requires a trivial computational cost.

Thus, we decided to use the objective function value of the best solution obtained during a search process as the quality measure and the total number of times of evaluating the objective function spent in finding this best solution as the efficiency measure. Given such an evaluation gauge, it is manifest that the number of times of evaluating the objective function is dependent on the number of search iterations and the value of the elite capacity. If, for example, the number of iterations is n , the elite capacity is e , and the number of eligible roadway subnetworks is r , the number of times of evaluating the objective function is of the order of $O(nr/e)$ during a search process.

We applied the algorithmic procedure for all the example networks with the following combinations of values of the search parameters: 1) *elite_capacity*: 1, 3, 5, 7, 9 and 11; 2) *tabu_tenure*: [3, 7], [8, 12] and [13, 17]; 3) *freq_threshold*: 0.80, 0.85, 0.90 and 0.95. Tabu tenure is a parameter that often shows a comparable performance among a number of its values in a given short range, so we confine the calibration for tabu tenure to a range rather than a precisely tuned single value. Many previous applications showed that it is not necessary and possible to find a specific tabu tenure value that works best for all instances of a problem class; instead, we expect that a randomly chosen number from a small range of tabu tenure, e.g., [8, 12], would offer a comparable search performance to other numbers in the same range.

The functionality of the residence frequency threshold is to adjust the magnitude and frequency of intensification during a tabu search process. Our experiments showed that a relatively low value of residence frequency threshold often results in local optimum traps, which indicates an overuse of intensification; while, in contrast, a very high frequency threshold is seldom satisfied with most lane-reversal elements, which results in an underutilization of intensification. In fact, by checking the results of a considerable set of experiments with each preset frequency threshold value and randomly selected tabu tenure and elite capacity values, we found that the residence frequency threshold is quite sensitive to the algorithm performance and a high threshold value is generally needed. Therefore, we quickly filtered out other frequency threshold values in the candidate set than $freq_threshold = 0.90$, where this latter value is used for the further calibration and implementation.

Given this specific $freq_threshold$ value, we applied the developed algorithmic procedure with each combination of a single $elite_capacity$ and a single $tabu_tenure$ value to all the given example networks repeatedly, which results in 35 experiments for each network. The calibration results for each set of calibration parameter values are presented in [Table 5.1](#), which are gauged by both the solution quality and search efficiency.

In these tables, the solution quality is simply denoted by the objective function value of the best solution (at the numerator position of each element) and search efficiency is gauged by the number of objective evaluations spent on finding the best solution (at the denominator position of each element) as well as the number of corresponding search iterations (in the parentheses at the denominator position of each element).

Table 5.1 The calibration results with regard to the elite capacity and tabu tenure

		Tabu tenure					
		[3, 7]		[8, 12]		[13, 17]	
Elite capacity	1	325,700	325,700	325,700	325,700	325,700	330,400
		355 (22)	398 (23)	396 (35)	374 (37)	461 (44)	250 (22)
	5	331,600	325,700	327,600	325,700	326,500	327,100
		482 (40)	496 (52)	578 (53)	548 (48)	811 (91)	711 (69)
	9	327,700	327,700	325,700	327,800	327,800	329,000
		194 (25)	282 (48)	391 (41)	333 (34)	369 (34)	332 (50)
	13	331,600	331,600	329,000	329,000	330,400	329,000
		287 (35)	318 (45)	358 (49)	290 (39)	305 (41)	351 (49)
	17	327,700	327,700	329,000	331,600	331,600	330,400
		257 (31)	235 (42)	294 (35)	210 (37)	240 (43)	269 (45)

(a) The calibration results of *elite_capacity* and *tabu_tenure* for example network 1

		Tabu tenure					
		[3, 7]		[8, 12]		[13, 17]	
Elite capacity	1	137,480	136,630	136,630	136,630	136,630	136,630
		664 (28)	954 (49)	776 (36)	776 (36)	772 (36)	775 (36)
	5	137,350	136,630	137,480	136,630	136,630	138,000
		231 (30)	450 (56)	390 (51)	401 (49)	416 (54)	235 (21)
	9	137,350	137,350	136,630	137,750	138,000	137,480
		159 (29)	347 (59)	316 (55)	331 (56)	106 (21)	224 (24)
	11	137,480	138,000	136,630	137,480	138,000	138,000
		138 (29)	86 (21)	283 (39)	248 (55)	352 (43)	86 (21)
	17	137,350	137,480	137,480	137,750	138,000	137,750
		152 (29)	134 (35)	223 (38)	189 (42)	178 (37)	116 (25)

(b) The calibration results of *elite_capacity* and *tabu_tenure* for example network 2

Table 5.1 (Continued)

		Tabu tenure					
		[3, 7]		[8, 12]		[13, 17]	
Elite capacity	1	275,630	269,700	269,700	268,470	274,010	274,010
		1,087 (35)	1,235 (48)	1,207 (48)	1,538 (58)	1,415 (53)	1,521 (59)
	5	276,540	276,540	268,470	268,470	279,100	279,680
		856 (62)	873 (70)	795 (65)	831 (68)	907 (70)	823 (53)
	9	274,010	276,540	270,490	269,700	274,010	279,100
		753 (58)	793 (65)	921 (78)	795 (69)	923 (72)	956 (65)
	11	279,680	269,700	278,700	278,700	282,430	278,700
		722 (42)	697 (61)	748 (53)	778 (62)	795 (65)	738 (58)
	17	279,680	282,430	279,680	278,700	275,630	282,430
		687 (45)	681 (55)	612 (45)	557 (48)	612 (58)	658 (58)

(c) The calibration results of *elite_capacity* and *tabu_tenure* for example network 3

		Tabu tenure					
		[3, 7]		[8, 12]		[13, 17]	
Elite capacity	1	5,573	5,472	5,472	5,472	5,577	5,573
		1,953 (50)	2,799 (74)	2,982 (80)	3,052 (82)	3,058 (82)	3,429 (93)
	5	5,573	5,541	5,573	5,472	5,604	5,654
		1,835 (94)	1,716 (91)	1,361 (70)	1,372 (75)	1,365 (72)	1,941 (99)
	9	5,514	5,472	5,541	5,481	5,514	5,541
		1,556 (95)	1,102 (62)	1,695 (98)	1,689 (92)	1,496 (84)	1,448 (85)
	13	5,472	5,614	5,527	5,564	5,614	5,553
		1,320 (77)	1,575 (84)	1,220 (64)	1,568 (83)	1,550 (80)	1,512 (87)
	17	5,573	5,541	5,472	5,564	5,472	5,553
		1,382 (88)	1,153 (73)	1,102 (62)	1,225 (56)	1,436 (99)	1,509 (87)

(d) The calibration results of *elite_capacity* and *tabu_tenure* for the Sioux Falls network

An overall interpretation and recommendation for the parameter settings can now be made in terms of the evaluation results of these example network problems under a variety of candidate parameter values (see [Table 5.1](#)). Tabu tenure may be the most concerned among all the parameters. The computational results showed that when $tabu_tenure \in [3, 7]$, a search is frequently trapped into a cycling state and cannot explore other promising feasible regions. Such a phenomenon reveals that this range of tabu tenure values may be too low to take advantage of the most important algorithmic feature of the tabu search technique, that is, to avoid local optima. On the other hand, when $tabu_tenure \in [13, 17]$, while the search heuristic performs quite well with good solutions, it often misses evidently better solutions that are close to its search itineraries, which may be caused by its overly strong tabu settings—some moves that can reach very good solutions are erroneously regarded as tabus during the search. It is found that for the given example problems the searches with $tabu_tenure \in [8, 12]$ typically find their best solutions by the most number of times and no single case of these searches encountered the cycling problem. Therefore, it is concluded that this tabu tenure range is an evidently better choice compared to other two candidate ranges, in which, particularly, $tabu_tenure = 12$ is the most attractive value.

However, the solution quality with regard to the elite capacity may not support a clear answer in interpreting its overall performance across the given example problem set. While no single elite capacity value shows a dominant preference among all the candidates, it is, as expected, that a smaller elite capacity tends to make the algorithm find the best solutions more frequently. It is not surprising that the conventional best-candidate-only rule, which is equivalent to $elite_capacity = 1$, conveys the best solutions most frequently among all the elite capacity numbers.

To maintain a consistent evaluation for the influences from the parameters on the solution quality, we standardized and synthesized the solution quality information from all the calibration experiments shown in [Figure 5.3](#), which, as we hope, can better visualize the quantitative correlation between the solution quality and the two important algorithmic parameters in a uniform and interactive manner. In this figure, each single objective value is calculated by applying the following standardization formula:

$$\begin{aligned} & \textit{standardized objective function value} \\ &= \frac{\textit{optimized objective function value} - \textit{best objective function value}}{\textit{worst objective function value} - \textit{best objective function value}} \end{aligned} \quad (5.1)$$

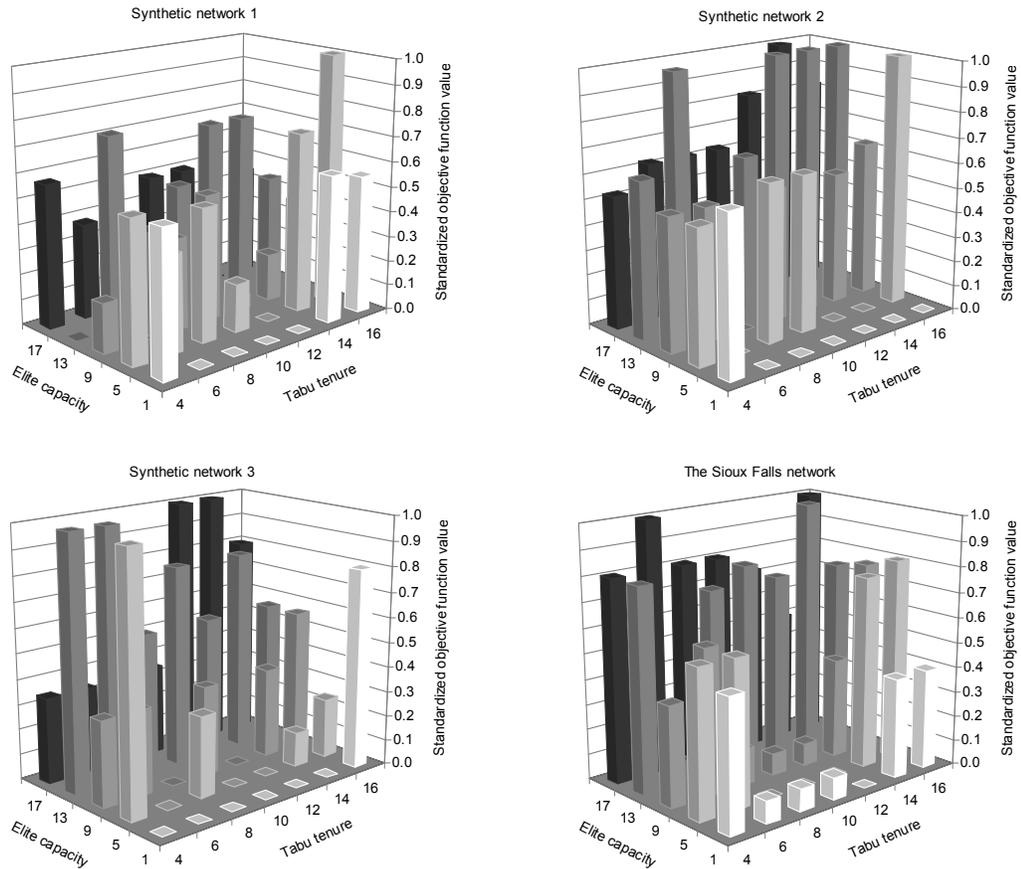
where the best and worst objective function values are lowest and highest ones from the set of the optimized objective function values under all the combinations of different algorithmic parameter values, i.e.,

$$\textit{best objective function value} = \min_{tt, ec, ft} \{\textit{optimized objective function values}\}, \quad (5.2)$$

$$\textit{worst objective function value} = \max_{tt, ec, ft} \{\textit{optimized objective function values}\}. \quad (5.3)$$

The overall relationship between the solution quality and the tabu tenure and elite capacity is depicted in [Figure 5.2\(a\)](#) for each problem case. Some of the information regarding the solution quality can be referred to in the preceding text. We also synthesized the information of the influences of the tabu tenure and elite capacity on the solution performance, respectively, in [Figure 5.2\(b\)](#) and [Figure 5.2\(c\)](#), which are

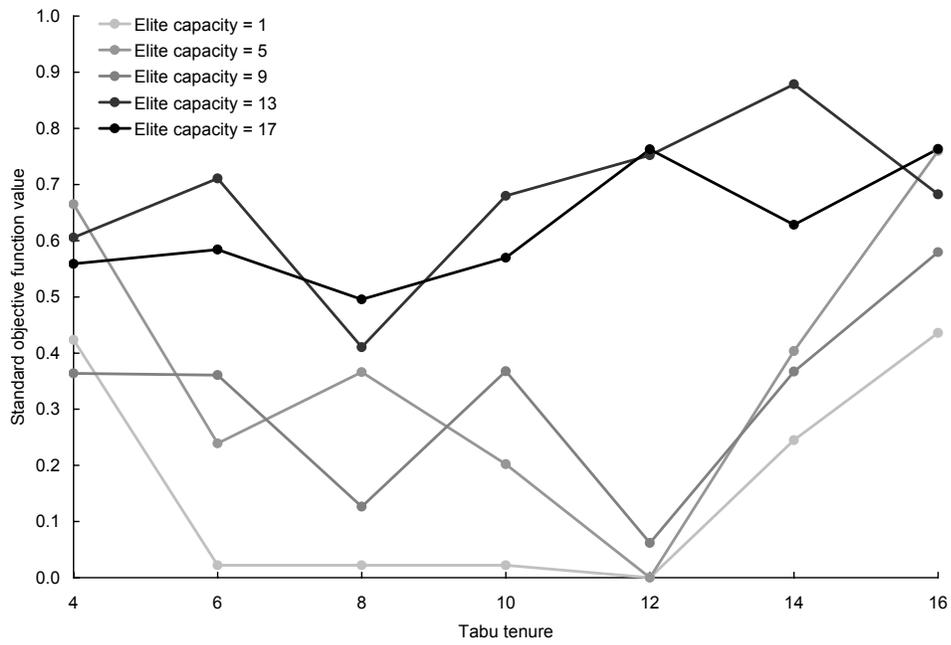
actually the projections of the 3-dimensional plot in Figure 5.2(a) to the planes of the tabu tenure and elite capacity.



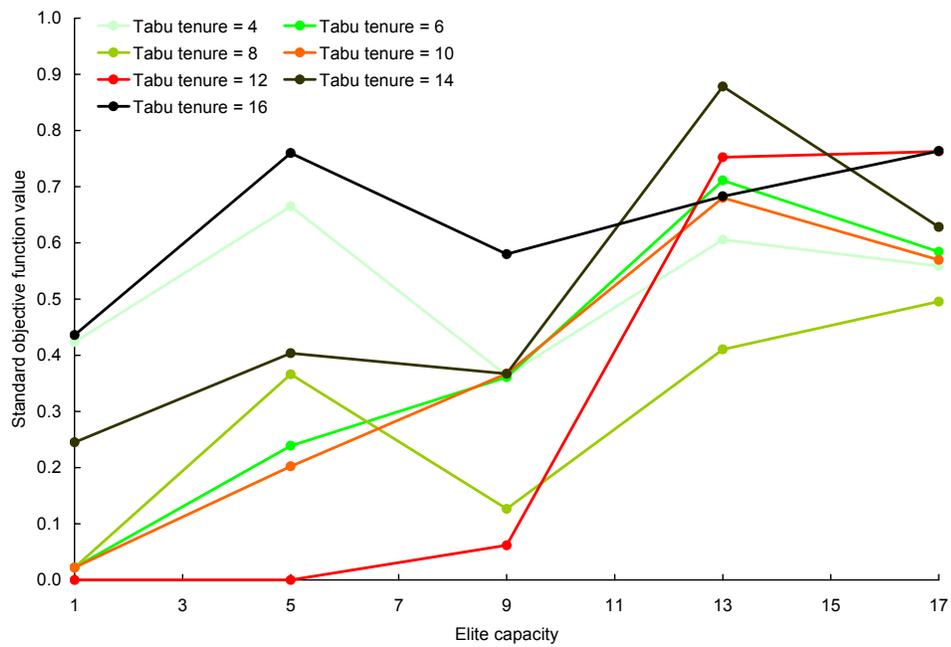
(a) Relationship between the solution quality and the tabu tenure and elite capacity

Figure 5.2 The solution quality and the tabu tenure and elite capacity

Figure 5.2 (Continued)



(b) Relationship between the solution quality and tabu tenure



(c) Relationship between the solution quality and elite capacity

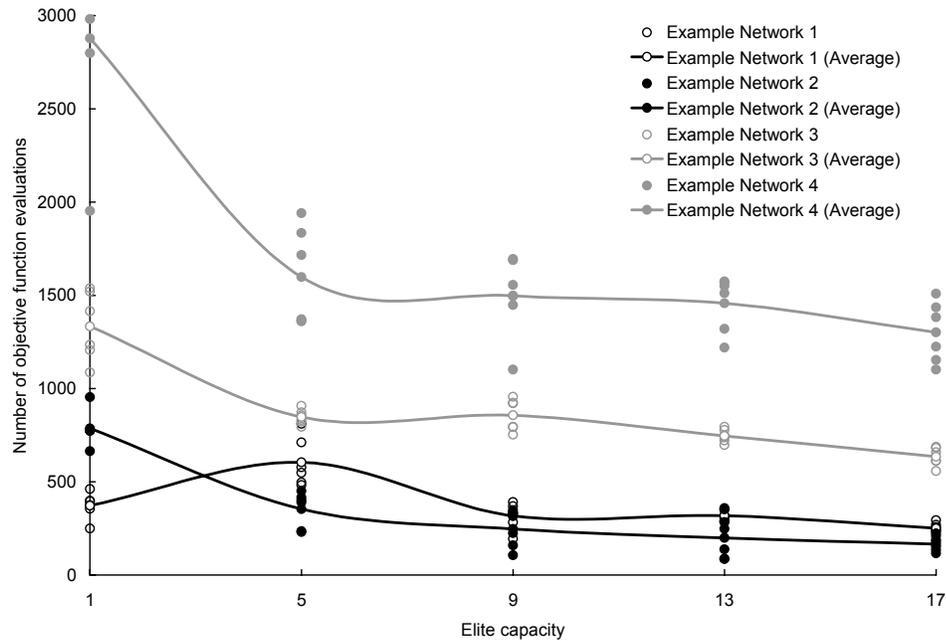


Figure 5.3 The computation efficiency and the elite capacity

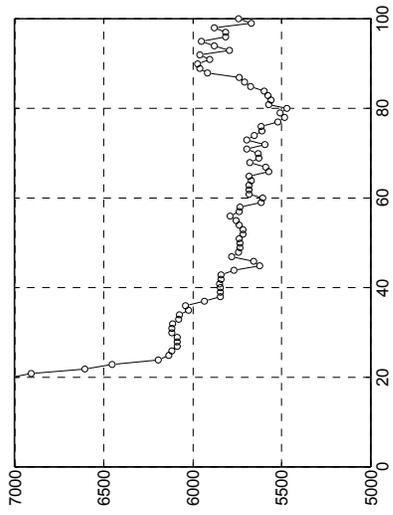
Some combinatorial impacts of the two algorithmic parameters on the solution quality can be extracted from the figures. These impacts can be more precisely described as follows. First, it is found that if the elite capacity is relatively small, such as, 1, 5 and 9, the search procedure finds the best solution or near-best solution when the tabu tenure is set as 8, 10 or 12, where the tabu tenure of 12 consistently delivers the best solution in all the cases. However, if the elite capacity is valued at a large number, such as, 13 and 17, the solution quality tends to be comparable among all the preset tabu tenure values. Second, a tabu tenure value in the intermediate range typically provides a lower objective function value, while the solution performance significantly worsens if the tabu tenure is too small (e.g., 4) or too large (e.g., 16).

On the other hand, the elite capacity imposes a considerable influence on the algorithm efficiency, which shows, in most of the situations, the computational cost required for attaining the best solution decreases, if an increasing elite capacity cost setting is given. One of the reasons is that the algorithm equipped with a larger elite capacity proceeds with iterations faster and hence is of greater potential in finding the best solutions early. We depicted the relationship between the computational cost and the elite capacity for all the example networks (see [Figure 5.3](#)), where the computational cost is counted by the number of evaluations spent on finding the lowest objective function value. These relationship curves clearly show that while an overall (approximately) decreasing relationship is observed over the whole range of the elite capacity values set in the experiments, the decreasing rate is much higher when elite capacity value is relatively small, say, $elite_capacity < 5 \sim 6$. Moreover, such a phenomenon is more apparent for example problems with larger size. The computational cost saving with a larger elite capacity value tends to be marginal when $elite_capacity \geq 7$. Considering the increasing risk in lowering the solution quality by a large elite capacity value, a choice of $elite_capacity = 7$ may provide the best compromise between the solution quality and computational efficiency.

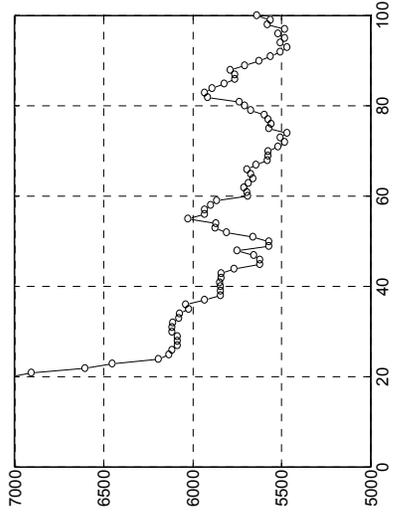
In terms of the above quantitative evaluation and analysis, we suggested the set of $tabu_tenure = 12$, $elite_capacity = 7$ and $freq_threshold = 0.90$ as the best choice of these parameter values for local searches. This chosen set of parameter values will be carried on when the algorithmic procedure is used to solve a larger evacuation network optimization problem.

The influence imposed by the algorithmic parameters on the searching behavior during a local search can be observed by tracking the search itinerary that is represented by a

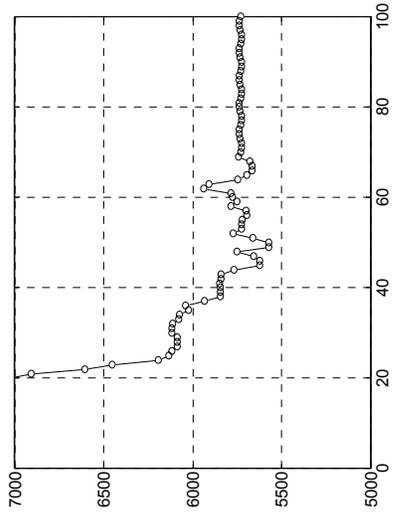
plot of the solution quality over the search iterations. As an illustration, we plotted such search itineraries of solving the Sioux Falls evacuation problem with different tabu tenure and elite capacity settings, where the frequency threshold for intensification, as suggested earlier, is set as 0.90 (see [Figure 5.4](#)). A glance on the solution itineraries for the Sioux Falls network shows us the search behavior along the iterations and the algorithmic capability of escaping local optima. A common feature within these itineraries is that the search process rapidly reaches at its first several local optima a solution close to the final optimized solution. In fact, we have observed the similar phenomena in other example networks. This result may be due to the relatively simple structure and small size of these networks. In an evacuation network, the closer a link is to an egress node, the quicker the lane assignment on this link can be determined during the search process. Specifically, for example, it can be quickly found by the search procedure that a link connecting to an egress node should be assigned with all its lanes towards to the egress node. Thus, in a network of relatively small size, the lane reversal directions of a large number of links that are close to the egress nodes may be determined at the initial stage of the search process, which constitute the most part of the final optimized solution. With the increasing size of a network and the decreasing number of egress nodes, we should expect that the search procedure overcome more local optima until reaching the optimized solution.



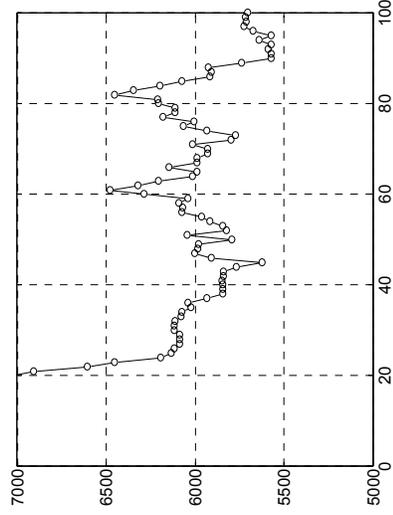
(a) elite_capacity = 1, tenure = 4



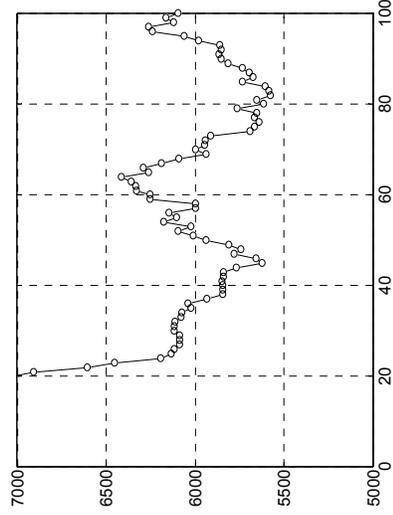
(b) elite_capacity = 1, tenure = 6



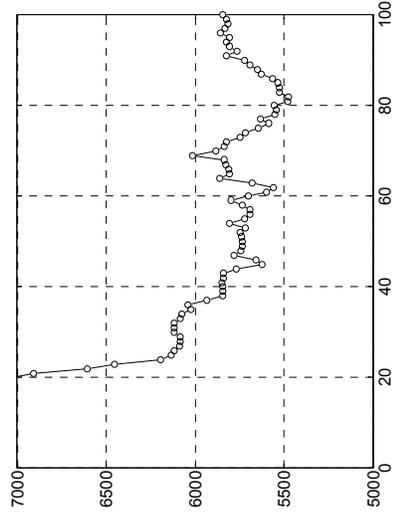
(c) elite_capacity = 1, tenure = 8



(d) elite_capacity = 1, tenure = 12



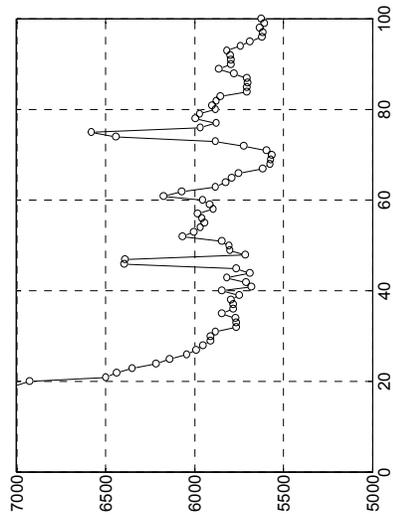
(e) elite_capacity = 1, tenure = 14



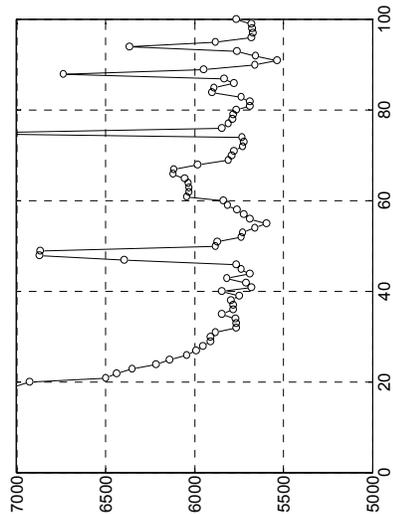
(f) elite_capacity = 1, tenure = 16

Figure 5.4 Example solution itineraries of the search procedure

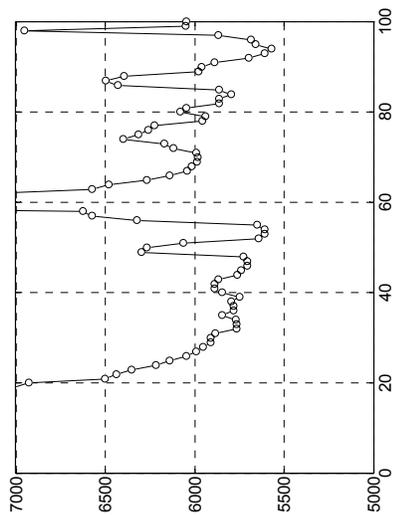
Figure 5.4 (Continued)



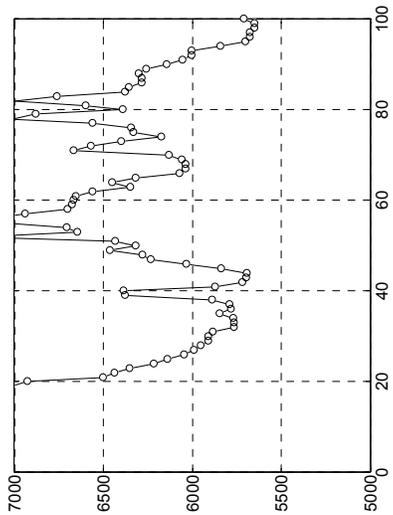
(i) elite_capacity = 5, tenure = 8



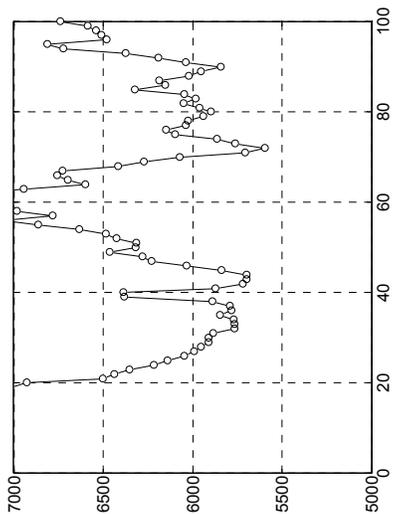
(h) elite_capacity = 5, tenure = 6



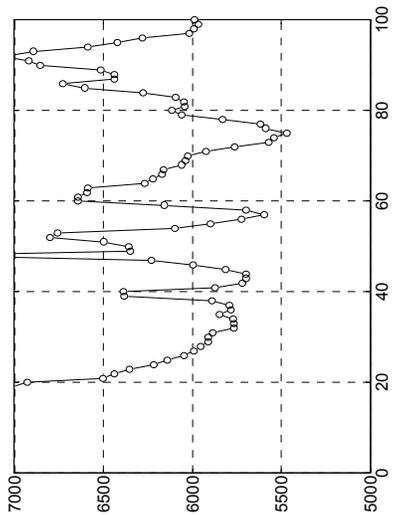
(g) elite_capacity = 5, tenure = 4



(l) elite_capacity = 5, tenure = 16

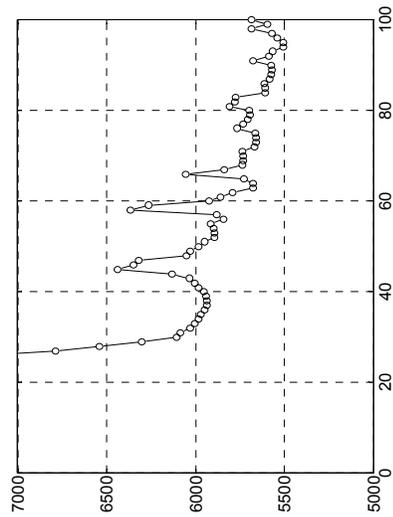


(k) elite_capacity = 5, tenure = 14

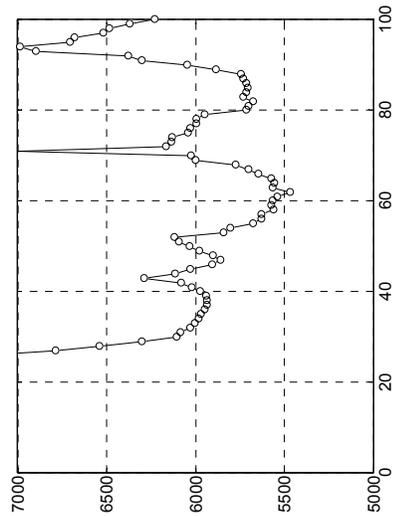


(j) elite_capacity = 5, tenure = 12

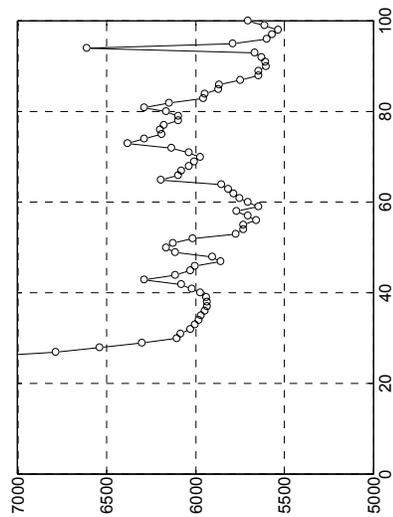
Figure 5.4 (Continued)



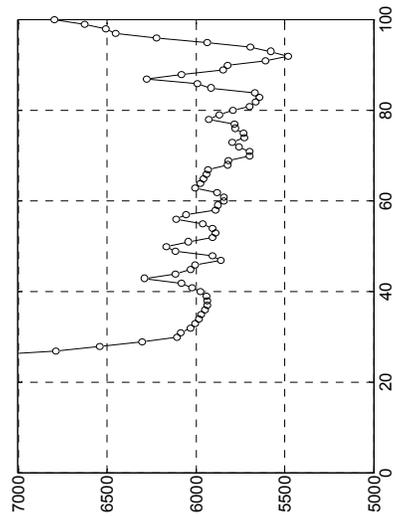
(m) elite_capacity = 9, tenure = 4



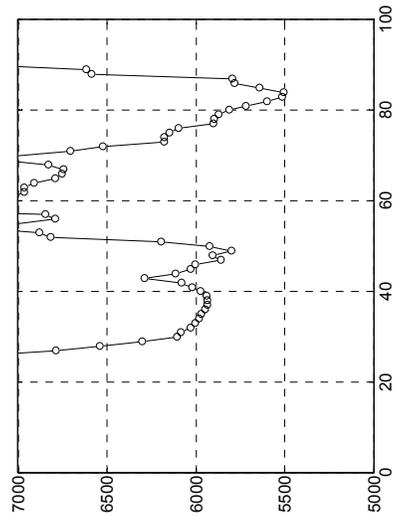
(n) elite_capacity = 9, tenure = 6



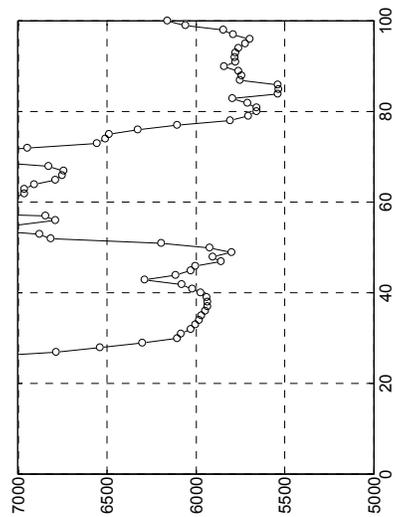
(o) elite_capacity = 9, tenure = 8



(p) elite_capacity = 9, tenure = 12

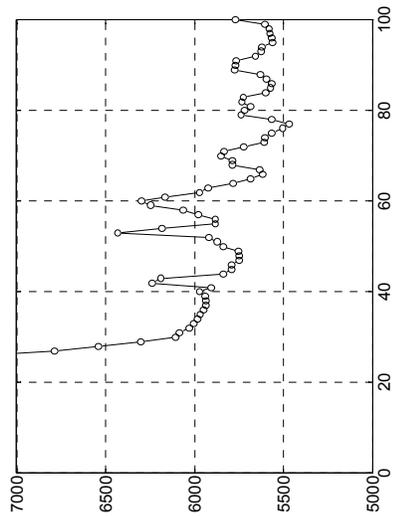


(q) elite_capacity = 9, tenure = 14

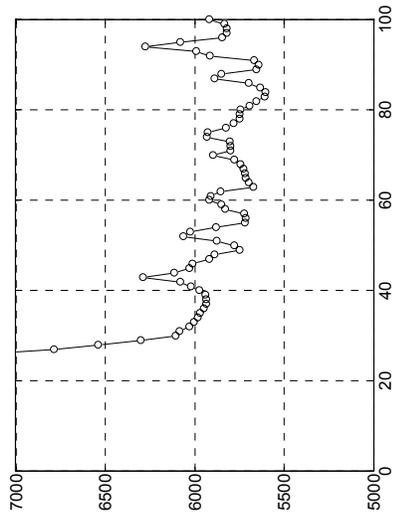


(r) elite_capacity = 9, tenure = 16

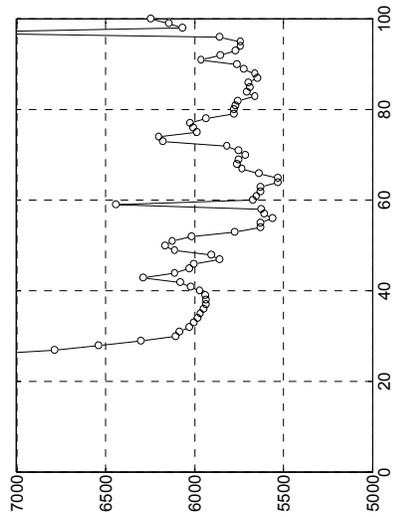
Figure 5.4 (Continued)



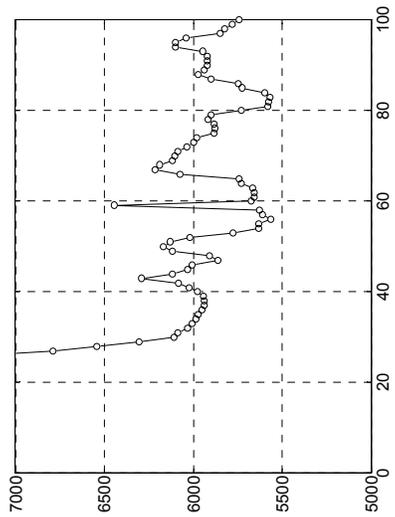
(s) elite_capacity = 13, tenure = 4



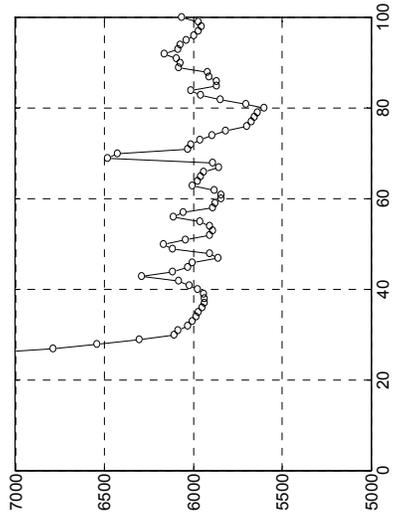
(t) elite_capacity = 13, tenure = 6



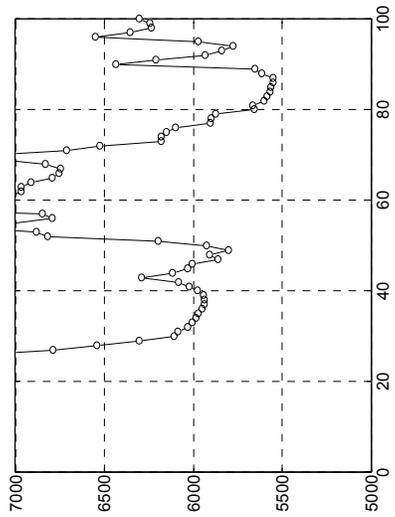
(u) elite_capacity = 13, tenure = 8



(v) elite_capacity = 13, tenure = 12

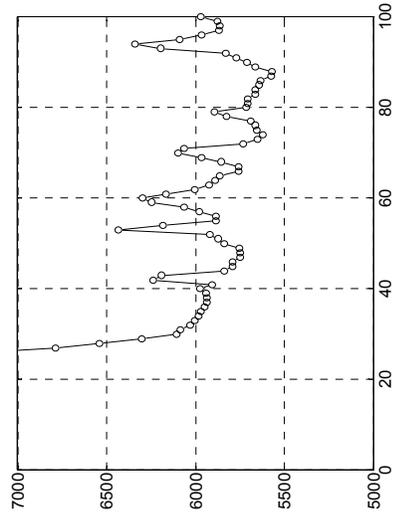


(w) elite_capacity = 13, tenure = 14

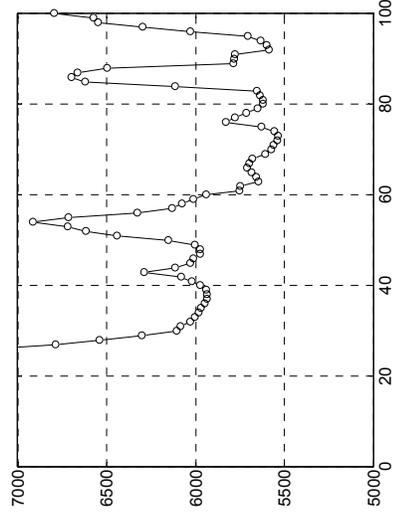


(x) elite_capacity = 13, tenure = 16

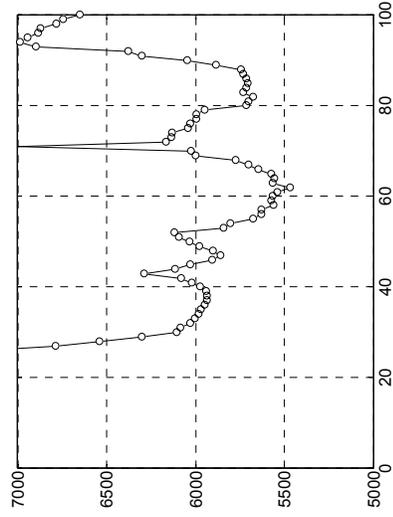
Figure 5.4 (Continued)



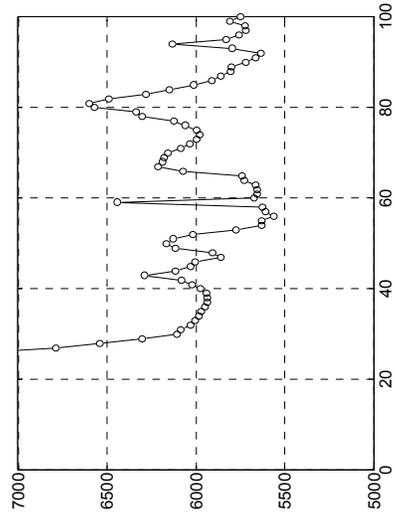
(y) elite_capacity = 17, tenure = 4



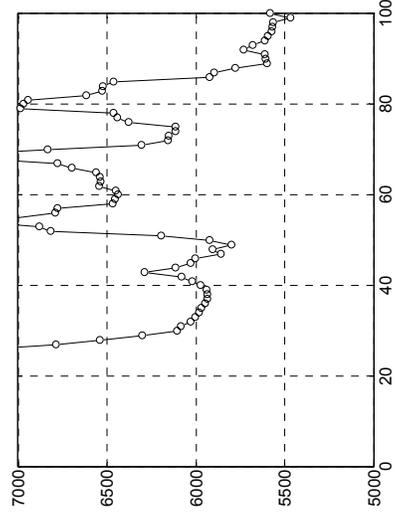
(z) elite_capacity = 17, tenure = 6



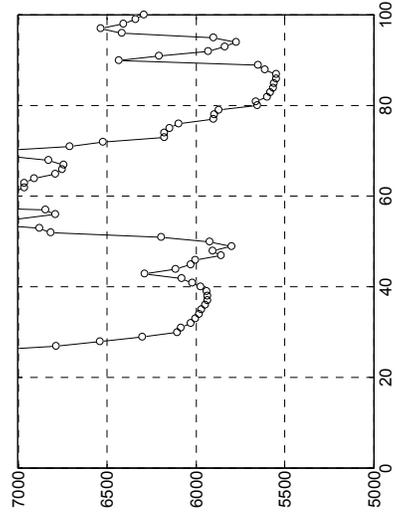
(aa) elite_capacity = 17, tenure = 8



(ab) elite_capacity = 17, tenure = 12



(ac) elite_capacity = 17, tenure = 14



(ad) elite_capacity = 17, tenure = 16

5.3 Diversification search calibration

For the diversification search, two more parameters need to be considered, which control a search process on a higher level, including the maximum number of non-improving iterations in a local search prior to starting a diversification, *max_iteration_num*, and the maximum number of diversification moves in a search, *max_diversification_num*. We examined the results from a large number of tests in the given four example networks with the preset range of local search parameters. It has been observed that a setting of *max_iteration_num* = 50 can satisfy the need for most of the evacuation network cases of the problem type defined here.

In many cases, diversification is an optional algorithmic element in a tabu search procedure. It does not necessarily improve the optimality condition of the final solution obtained after one or diversification transfers are conducted. Our computing experience suggested that while diversification may or may not improve the best network solution, *max_diversification_num* = 3 may be an appropriate number to invoke diversification.

5.4 Performance evaluation

Given all the computational results, this section is intended to provide a summary of the computational performance of the solution procedure and characterize the optimized network solutions of the set of example problems. It is our hope that the latter task could provide us with some general empirical insights for devising more efficient heuristic rules if part of the optimal solution pattern could be reasonably predicted.

A deficiency with the solution evaluation in this research is that there is so far no exact solution method available to solve the defined network optimization problem and to further provide an optimality criterion for the approximate solutions. This deficiency is not only due to the computational obstacle but also the methodological lack. For the first two small examples, however, it can be confirmed by using the simple though tedious exhaustive method that their best solutions attained by the proposed heuristic method are the truly optimal solutions. As another attempt of comparing the solution quality, we examined in all the examples the gaps between the best and worst solutions resulted from the whole population of calibration parameter values. It has been found that the average gap is around 1.1 percent and the largest gap is 4.2 percent. This result is rather promising, since even if a non-optimal parameter setting is chosen, one could still expect to obtain a high-quality solution from using this heuristic procedure. From the computational perspective, this finding is important and meaningful, because, due to the computing resource and time limits, it is not possible to conduct a large number of experiments with a broad range of problem types, sizes and parameter settings. In this regard, we can conclude in an empirical manner that the developed algorithmic procedure is a rather robust optimization heuristic.

Although there is some uncertainty within the optimality conditions provided by the heuristic solutions, it would be interesting to estimate the network efficiency benefits gained from implementing an optimized network solution compared to the original network configuration, which provides an alternative perspective for evaluating the goodness of a heuristic solution. In [Figure 5.5](#), the best and average objective function values of the optimized solutions obtained by the solution procedure are compared to the ones associated with the original network configurations.

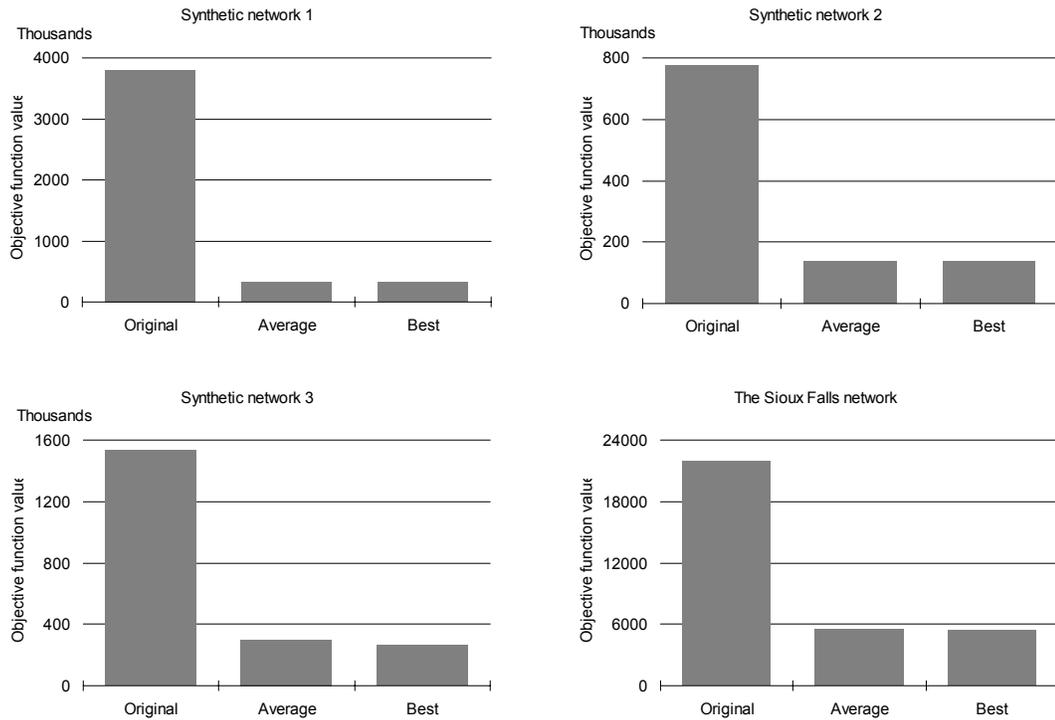


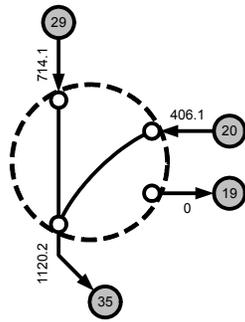
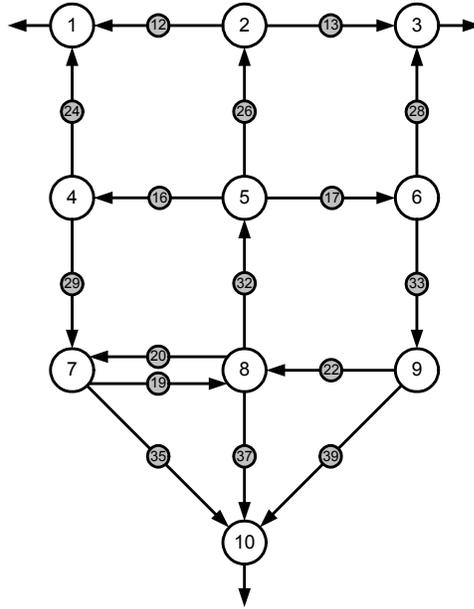
Figure 5.5 Comparison of the objective function values of the optimized solutions and the original configurations

The magnitude of the network efficiency improvement by the optimization process can be measured by comparing the objective function values affiliated with the optimized solutions and the original network configurations. It can be seen from [Figure 5.5](#) that a significant reduction from the objective function values of the original configurations is made by the optimized solutions procedure, where the reduction ranges approximately from 75 to 90 percent in these example problems. Furthermore, for any problem case, the average objective function value of all the heuristic solutions seems very close to the best solution in terms of the numerical scale given in the figure, which, once again, enhances our confidence on the optimality (or

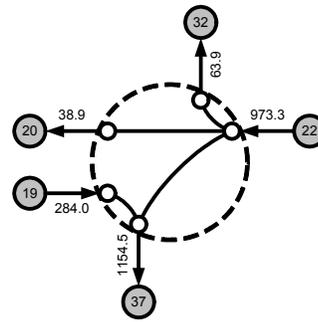
near-optimality) robustness of the developed metaheuristic procedure, in that its solution performance is relatively insensitive to the parameter settings.

The original configurations of these example networks have been all set to mimic the most common two-way street settings in typical urban traffic area, for which it is presumed that such a traffic network is designed to provide the capacity and connectivity for the daily commuting traffic. The optimized network solutions should, as expected, have a considerably different topological structure from their original network configurations, due to the evacuation demand pattern that is drastically different from the commuting demand pattern. To evaluate the resulting topological difference, we depict in the following figure ([Figure 5.6](#)) the network configuration of the best solution of each example problem in its reduced form. It is well known that at any intersection a crossing-free traffic movement configuration can be always guaranteed if all the approaching links to this intersection are one-way links. However, if this is not the case, the existence of crossing points would be not only dependent on the adjacent lane reversal configuration but also the traffic flow pattern. To present the complete solutions, along each optimized network, we also magnified the traffic flow movements at each intersection subnetwork connecting with at least one two-way link.

While capacity and connectivity variations are shown within the optimized network solutions of these example problems, some common topological features of the roadway section and intersection configurations can be summarized as follows.



Node 7 intersection subnetwork

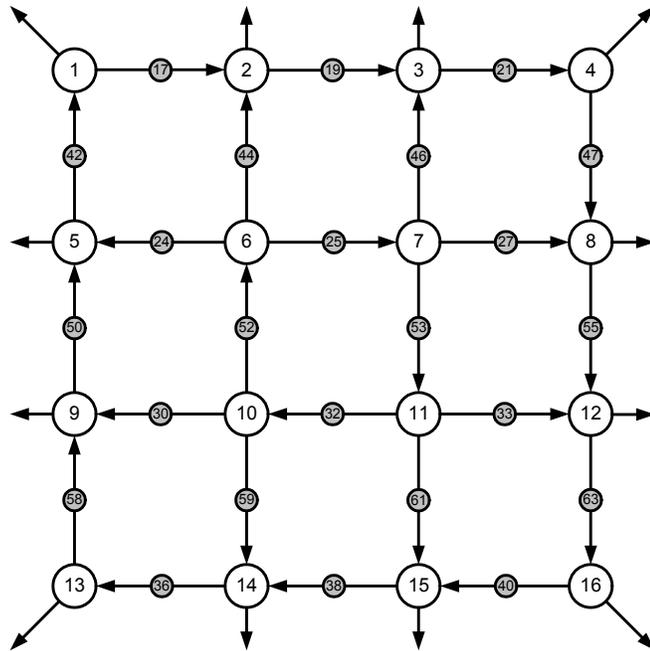


Node 8 intersection subnetwork

(a) The optimized solution of example network 1 (26 nodes, 32 links)

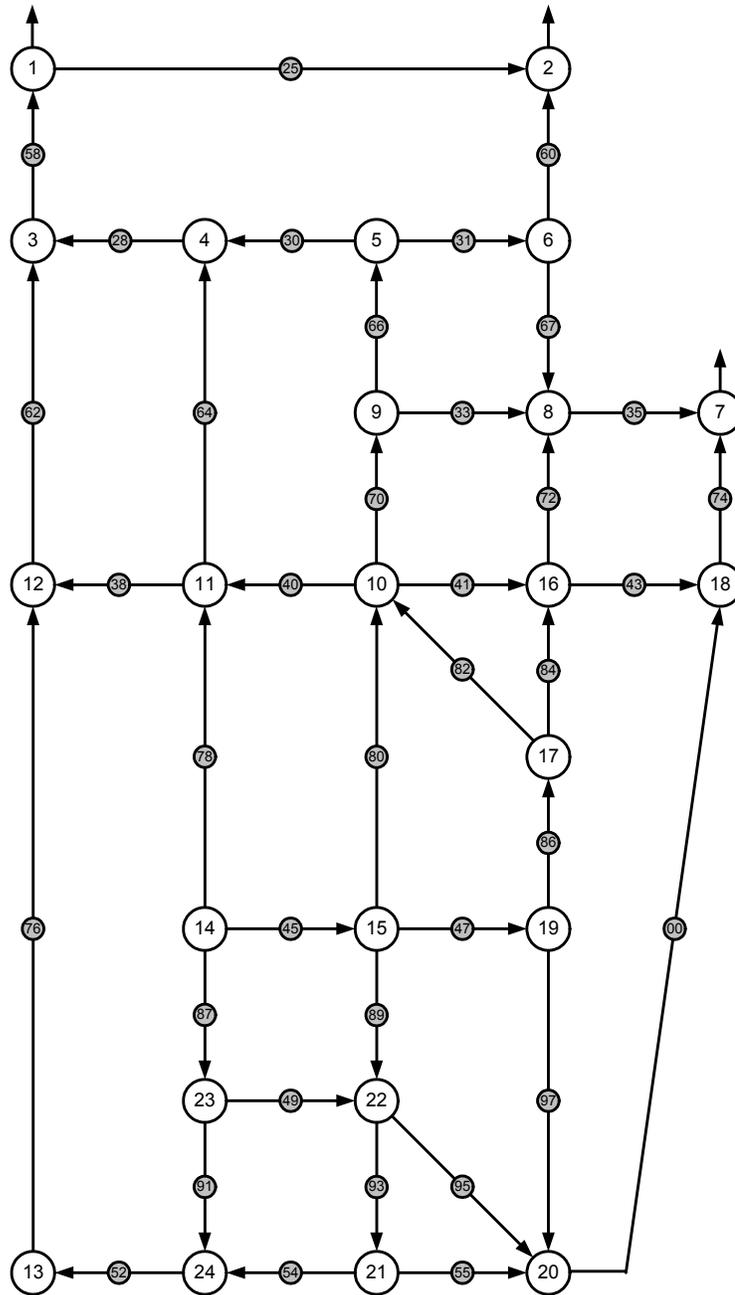
Figure 5.6 The optimized solutions of the example networks

Figure 5.6 (Continued)



(b) The optimized solution of example network 2 (40 nodes, 48 links)

Figure 5.6 (Continued)



(d) The optimized solution of the Sioux Falls network (62 nodes, 76 links)

Among all the network solutions, there are few two-way links, especially in the second synthetic network and the Sioux Falls network, in which all links are fully reversed. This reflects the need of reversing as many lanes as possible for the outbound traffic and also contributes to the requirement of crossing elimination. An intersection with all one-way approaching links must be a crossing-elimination intersection subnetwork. This phenomenon suggests that a problem with the full lane reversal settings may be a good approximate to the integrated evacuation planning problem defined in this study. Such an alternative model with full lane reversals is attractive in that it has fewer discrete decision variables and can be solved more efficiently.

It is found that along with the change from two-way networks (i.e., the original configurations) to one-way networks (i.e., the optimized solutions), the numbers of nodes and links are significantly reduced. If a two-way roadway section is converted to a one-way section, the numbers of nodes and links are reduced by 1 and 2, respectively, and the number of intersection links adjacent to this roadway section could be reduced by up to 6. As an example, in the reduced form of the Sioux Falls network, the numbers of nodes and links in its original configuration are 100 and 152, respectively; the numbers are decreased to 62 and 76 in its optimized solution. This problem size reduction greatly favors the estimation of the network flow pattern during the optimization process. In fact, it has been observed that the search speed accelerates significantly after a certain number of iterations from the initial solution.

Depending on the network topology and destination distribution, the lane reversal configurations in these optimized network solutions are of different spatial patterns. However, all these configurations appear with an approximate destination-oriented feature. Most of the reversed links are assigned with a traffic direction pointing to a

closest exit node, especially those links in the vicinity of an exit node. This feature is consistent with our previous engineering judgments in devising contraflow plans.

Concurrently, it should be noted that there are also a number of other roadway sections with the optimized contraflow directions that may not accord with our intuition or give an intuitive answer. Most of such sections are located at the places where there are multiple competing exit nodes or the links to be reversed are relatively far from any exit node. To illustrate a few examples, see in the Sioux Falls network the links between nodes 4 and 5, between nodes 5 and 9, and between 14 and 15. Without a systematic optimization method like the solution procedure presented here, it may be difficult to identify the optimal lane reversal configurations for these roadway sections, especially for those networks with large size.

Given the above computation results, it is also necessary to briefly comment on the performance of the developed LR-TS algorithmic procedure. Due to the heuristic nature, the search behavior and efficiency of the procedure may not be accurately predicted. In our experience with the limited number of experiments, it has been found that the required computational cost (i.e., the number of objective function evaluations) increases roughly at a polynomial rate with the problem size (i.e., the numbers of nodes and links). In this regard, the number of candidate lane-reversal roadway sections and crossing-elimination intersections are the primary factors. The former determines how many evaluations are needed at each iteration and how many iterations are expected during the whole search process. The latter determines the times of solving the intersection traffic movement optimization subproblem at each evaluation.

It has been realized that there is no feasible way so far to precisely assess the solution quality of the evacuation network optimization problem of the type defined here, except those simplest cases that could be solved by the exhaustive enumeration. However, the experiments of the example problems under a variety of demand and supply settings as well as a broad range of parameter settings showed that the solution procedure is a rather robust optimization method that can at least find near-optimal solutions.

Although a set of heuristic rules embedded in the solution procedure are intentionally designed to accelerate the search course, it is still a computationally intensive process, even for medium-size problems. Two algorithmic components of the procedure may let us take advantage of parallel computing as a very attractive computing mechanism in implementing the algorithm. The two components that could be implemented in a parallel computing form are respectively the evaluations of candidate network solutions during an iteration and the examinations of traffic movement configurations at eligible intersections, where in each of the components there are a large number of separate and parallel optimization problems. As we discussed before, these optimization problems actually constitute the bottleneck of the heuristic search process. In this regard, parallel computing provides a natural way to reduce the computing time and the implementation could be realized in quite easily.

CHAPTER 6

EVACUATION PLANNING FOR A NUCLEAR POWER PLANT

What is rational is actual and what is actual is rational.
—G.W.F. Hegel

An evacuation planning case study is introduced in the following to demonstrate the capability and effectiveness of the models and solution procedures developed in this study. The assumed evacuation situation in this example comes from a nuclear power plant located in Monticello, Minnesota. Though, as we will see, some specific problem characteristics and model elements are set for particular needs arising from this target problem, our methodology, with some minor modification or adaptation if needed, can be applied to model and optimize evacuation networks under a variety of other emergency situations.

The objectives and settings of this Monticello case study enable us to investigate many modeling assumptions and insights and to assess the applicability and performance of our methodology. Two problem instances are formed and their corresponding solutions are accordingly developed and discussed. The first instance is an application of the basic model for searching for the optimal lane-reversal and crossing-elimination configurations in the Monticello evacuation network given different network supply settings; the second one is an application of the extended model that simultaneously optimizes the joint evacuation network optimization and emergency vehicle assignment scenario.

6.1 The evacuation network setting

Monticello, Minnesota, is a community of about 11,000 people located at the northern edge of Wright County, along the Mississippi River about 30 miles northwest of Minneapolis. The Monticello nuclear plant is owned by Northern States Power, a subsidiary of Xcel Energy, and is operated by Nuclear Management Company. The plant began operation in 1971, and is currently licensed until 2030. As enacted by the NRC and FEMA, an emergency planning zone (EPZ) with a 10-mile radius must be delimited centered at the site of any nuclear power plant in the U.S. Because the Monticello plant is along a river that forms a boundary between counties, the EPZ in this case covers areas in both Wright County and Sherburne County. [Figure 6.1](#) shows the general location of the plant and the 10-mile radius EPZ.

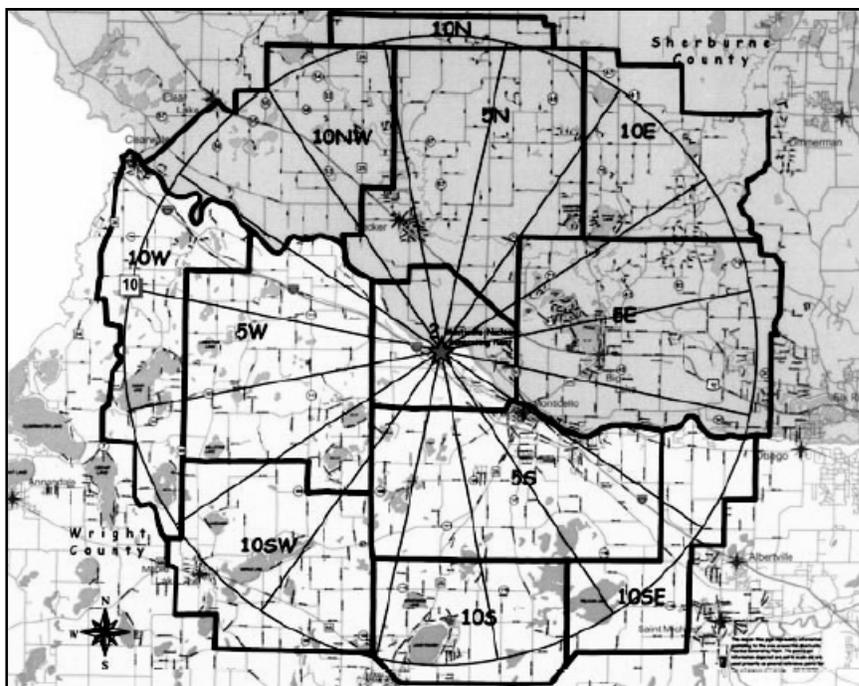


Figure 6.1 The emergency planning zone for the Monticello nuclear power plant

If a nuclear accident alarm is triggered, all inhabitants in the EPZ are required to leave the area so as to avoid potential exposure to a released radioactive plume. For evacuation planning purposes, an evacuation network is extracted from the regional surface street and highway network. The resulting evacuation network is shown in Figure 6.2.

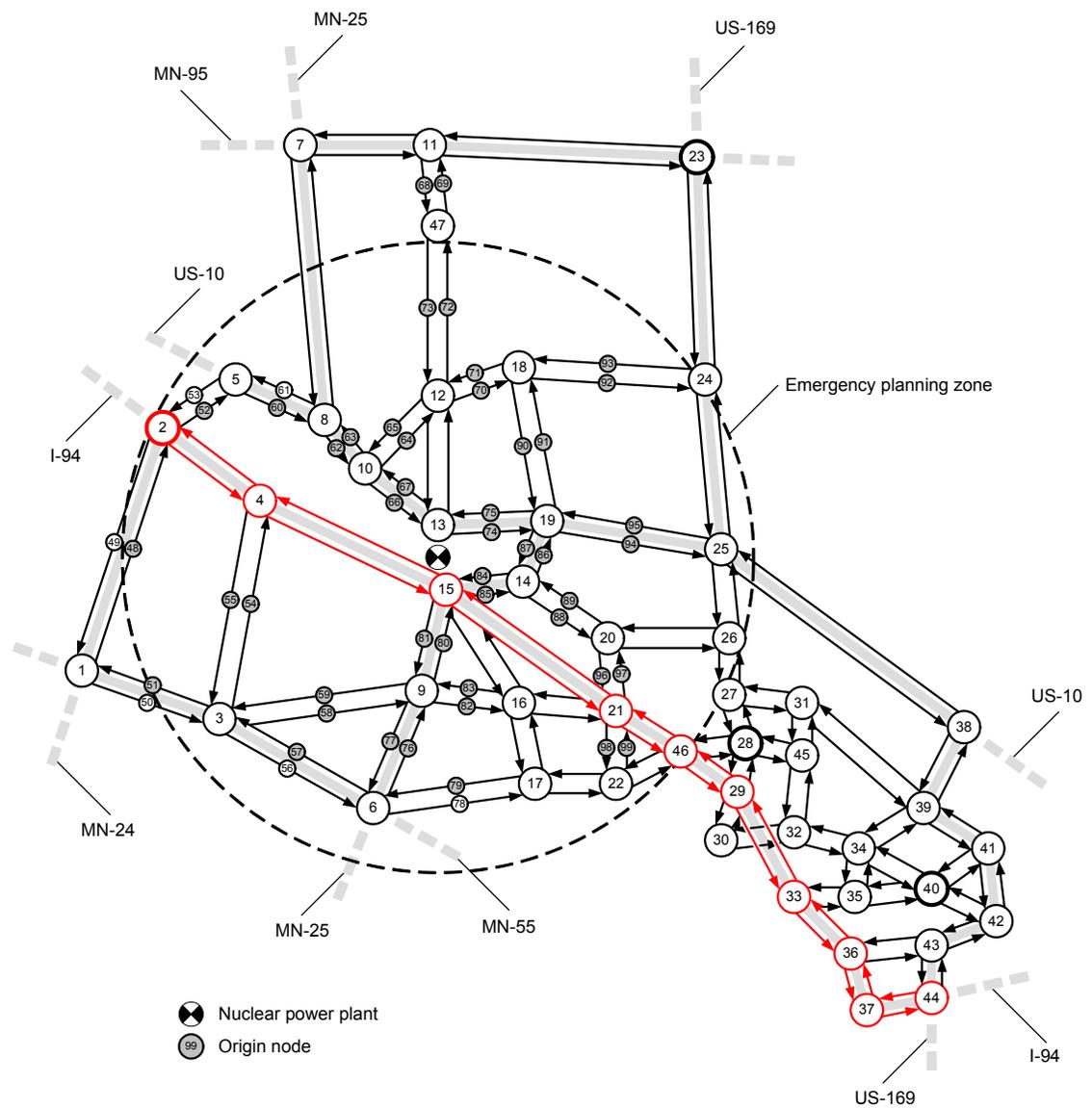


Figure 6.2 The Monticello evacuation network

There are 14 cities or towns in the network surrounding the nuclear power plant. The evacuation demand generation is estimated based on the demographic data of the region from the U.S. Census 2000 survey. The total number of evacuees from the network is about 42,000. The surge demand rates are approximated by using the historical diurnal curves of evacuation demand generation such as the ones shown in [Figure 3.3](#).

The Monticello evacuation region covers both urban and rural areas and the roadway system consists of both freeway and arterial segments. An interstate highway, I-94, spans the network (as denoted by a node chain, 2↔4↔15↔21↔46↔29↔33↔36↔37↔44, in [Figure 6.2](#)), while most of the remaining roadway segments are U.S. and state highways and regional arterials. These major arterials are also expected to serve as important arteries the evacuating traffic flow. These major arterial routes are highlighted in [Figure 6.2](#), including U.S. routes, US-10 and US-169, and state routes, MN-95, MN-25, MN-55 and MN-24. The nuclear power plant is located in the middle the EPZ, which is adjacent to node 15. Note that the interstate highway I-94 is the only uninterrupted traffic facility in the network and its capacity is significantly larger than other arterial roadways, so it is anticipated that this traffic corridor would be the most important evacuation pipeline and convey a large amount of traffic during the evacuation period.

The size of this network (i.e., 99 nodes and 200 links) may be relatively small compared to a regional or statewide evacuation network under some other emergency situations. However, given the fact that all the roadway segments of the network are

included as eligible contraflow segments, this network optimization problem still poses a very challenging computational task.

There have been a number of potential destination nodes identified for the Monticello evacuation network, including nodes 40, 28, 23 and 2 (refer to [Figure 6.2](#)). The first three destinations are all designated emergency reception centers located at local high schools: Osseo Junior High School (at node 40), Rogers High School (at node 28), and Princeton High School (at node 23). These reception centers can provide evacuees with basic accommodation facilities and medical services. The last destination node is supplemented as an additional egress, given that all the above reception centers are located in the east part of the area and may not attract those inhabitants residing in the west part. If residents in the west are guided or forced to evacuate toward any of the three reception centers, they may have to travel through some roadway segments in the proximity of the nuclear power plant. An evacuation plan with such a destination setting may result in some safety concern and psychological fear within the evacuating population. From a practical point of view, the I-94 westbound naturally provides a more convenient and accessible exit for those residents under a nuclear emergency situation. This leads us to suggest employing node 2 as an egress for the Monticello evacuation network if needed.

Among these potential evacuation destinations, the Osseo reception center has long been designated by the Sherburne County Sheriff's Department[†]; the addition of the Rogers and Princeton reception centers has been recently suggested by Nuclear

[†] Source: Monticello Nuclear Power Plant Emergency Planning Guide, accessible at http://www.co.sherburne.mn.us/sheriff/services/mnp_evacuation.htm.

Management Company (NMC)[‡], the plant operator. As for the I-94 west egress, its incorporation is the result of our preliminary spatial network analysis. No accommodation capacity requirement at reception centers has been estimated in the previous evacuation plans. In view of these varying egress availabilities, three destination settings, i.e., destination node 40 only, destination nodes 40, 28 and 23, and destinations 40, 28, 23 and 2, are suggested and investigated in this study. This resulted in three different network scenarios: the first scenario has only one egress (i.e., node 40) in the east part of the network; the second scenario has three egresses (i.e., nodes 40, 28 and 23) in the east and northeast; the third scenario includes all the egresses (i.e., nodes 40, 28, 23 and 2).

The traffic crossing pattern described in [Chapter 3](#) (see [Figure 3.1](#)) arises only from a standard four-leg intersection. Other traffic-crossing geometric settings, for example, a three-leg intersection or an at-grade interchange, have different crossing configurations and optimality conditions on traffic turning movements. However, both of these alternative traffic crossing roadway components appear in the Monticello network. Although these alternative geometric designs pose different traffic crossing situations, the lane-based network modeling principle described earlier can still be applied here to define their corresponding crossing-elimination constraints. In fact, the set of crossing-elimination constraints for these alternative traffic crossing cases can be readily specified, according to a set of similar rules to those applied to a four-leg intersection. As an example, we illustrated in [Figure 6.3](#) the traffic crossing patterns at a three-leg intersection and a diamond interchange.

[‡] Source: Monticello Nuclear Generating Plant Emergency Planning Guide and Calendar, accessible at http://www.nmcco.com/about_us/emergency/monticello/monti_home.htm.

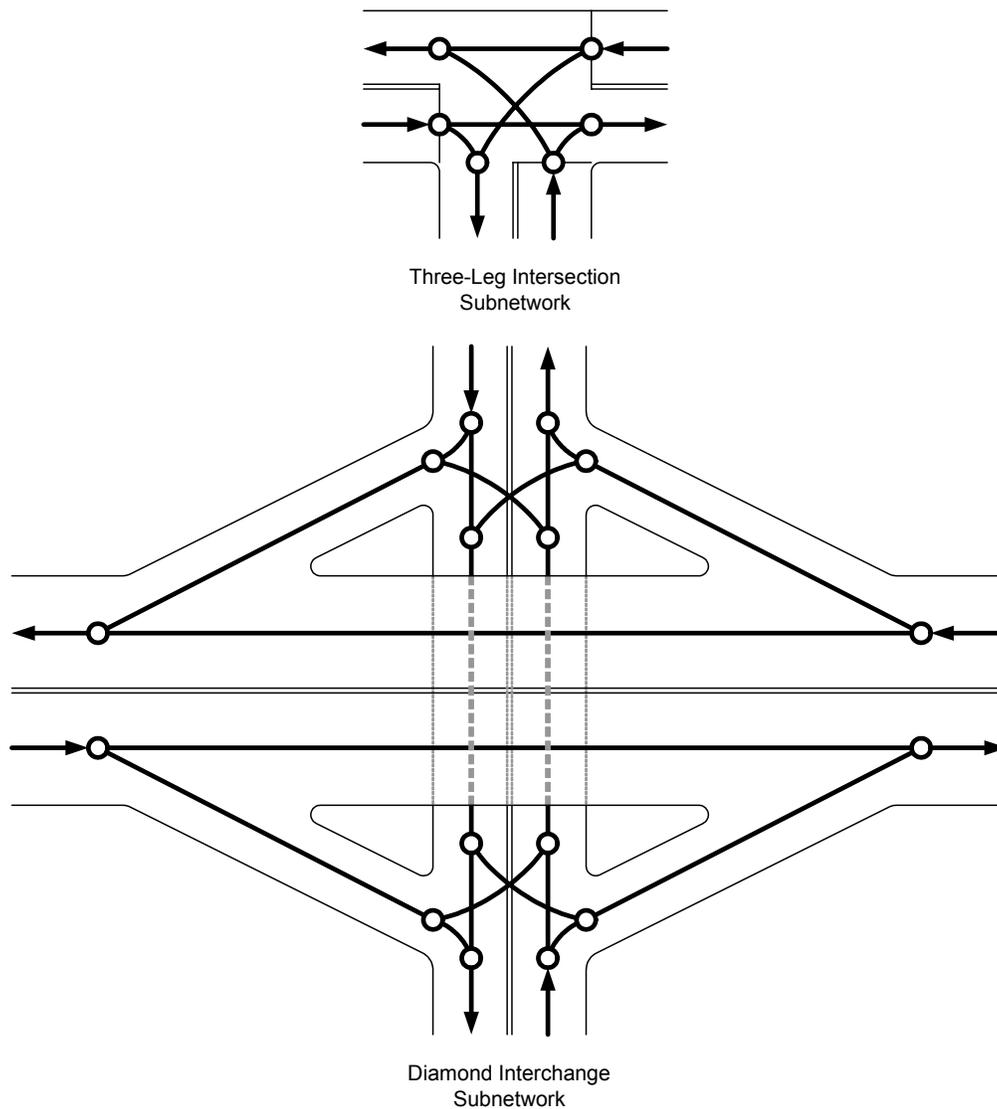


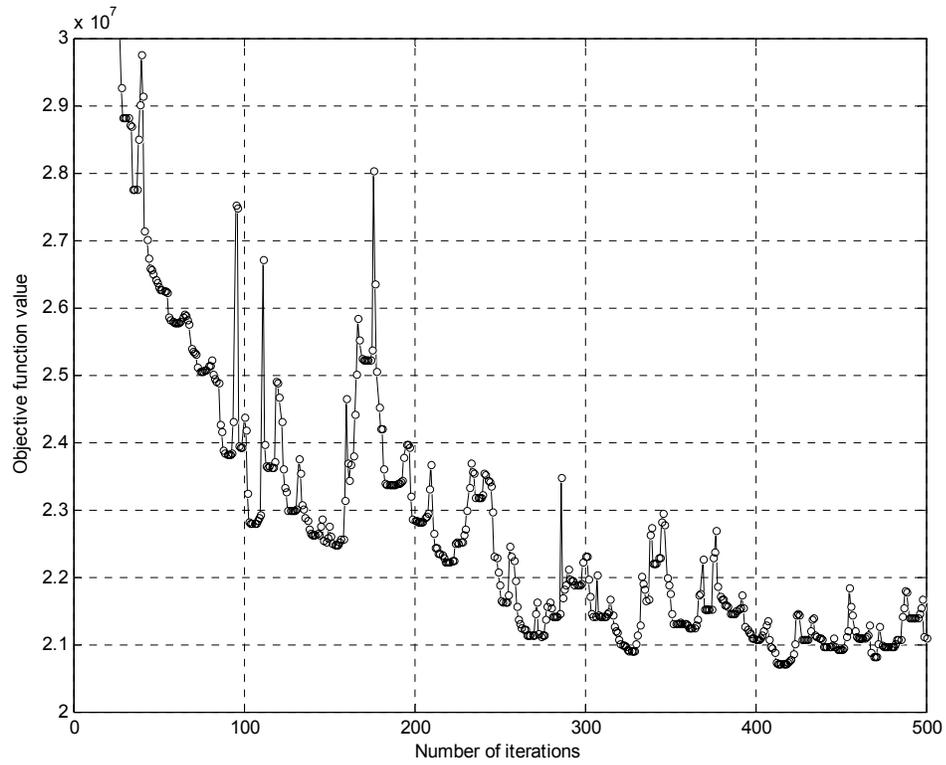
Figure 6.3 Traffic crossing points of alternative intersection/interchange designs

6.2 Evacuation network solution development and result analysis

The LR-TS solution procedure with the calibrated parameter set is then used to search the optimal evacuation plans for the three defined evacuation scenarios from the Monticello network. We implemented the algorithm in MATLAB 7.1 (R14) and

conducted all the experiments on a Windows-based PC with an Intel Pentium Dual-Core 1.80GHz CPU and 1MB memory. The resulting search itineraries and network solutions are presented in [Figure 6.4](#) and [Figure 6.5](#), respectively. The computational complexity with this set of evacuation network optimization problems can be seen from the search itineraries, in that a larger number of local optima are encountered during the search process than that of those small-size example networks used in the calibration. The search process requires more iterations to reach the optimal (or near-optimal) solutions, the solution procedure resorts to a larger number of objective function evaluations for each iteration, and each evaluation costs a longer computing time. Specifically, the optimized solutions of the three network scenarios with one, three and four egresses are identified at iteration 413, 479 and 334, and the computing times for finding these optimized solutions are 3.023×10^4 , 3.484×10^4 and 2.298×10^4 sec, respectively. The four-egress network scenario is least congested among the three due to its largest number of egresses and accordingly it reaches the optimized solution relatively faster.

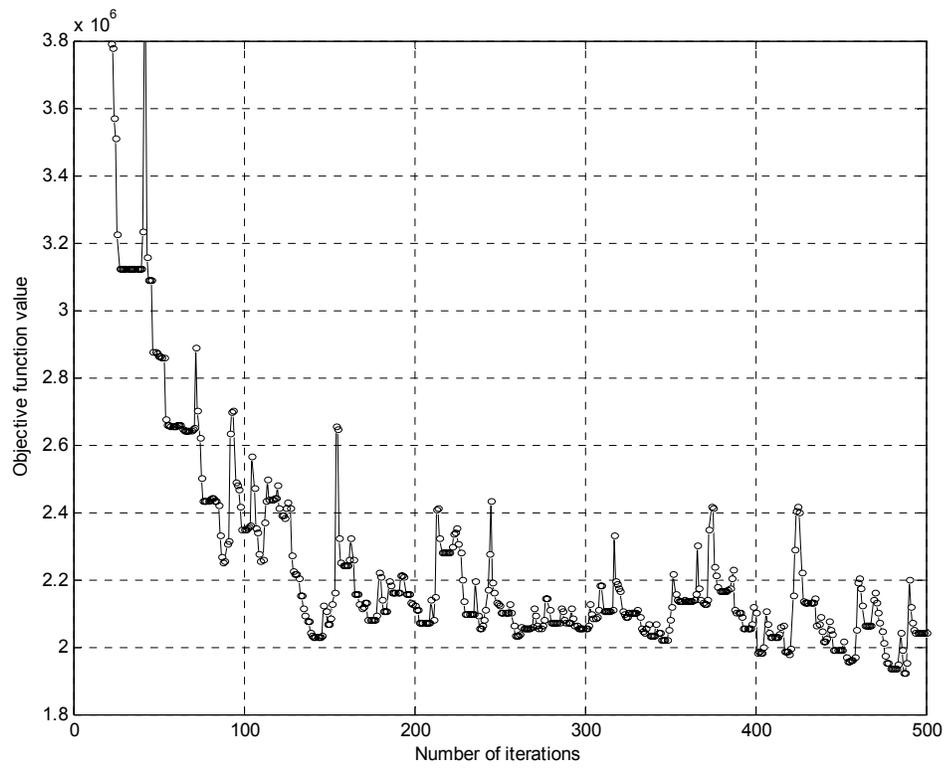
The lane-reversal directions in each of the network solutions constitute a destination-oriented pattern. This spatial characteristic can be described as follows, that is, emanating from the heart area of the network (which is far from any egress node), most roadway segments are reversed in such a way that the traffic is distributed over the network, moving outbound and merging at the egress nodes. Because of this, two-way roadway segments only exist near the center of the optimized evacuation network, where links are far away from any egress node and there may be no single contraflow direction for those links that can make the traffic distributed over the network more efficiently than the two-way traffic alignment. In other words, the traffic in the central area may be attracted by multiple destinations.



(a) Search itinerary for the Monticello network with one egress

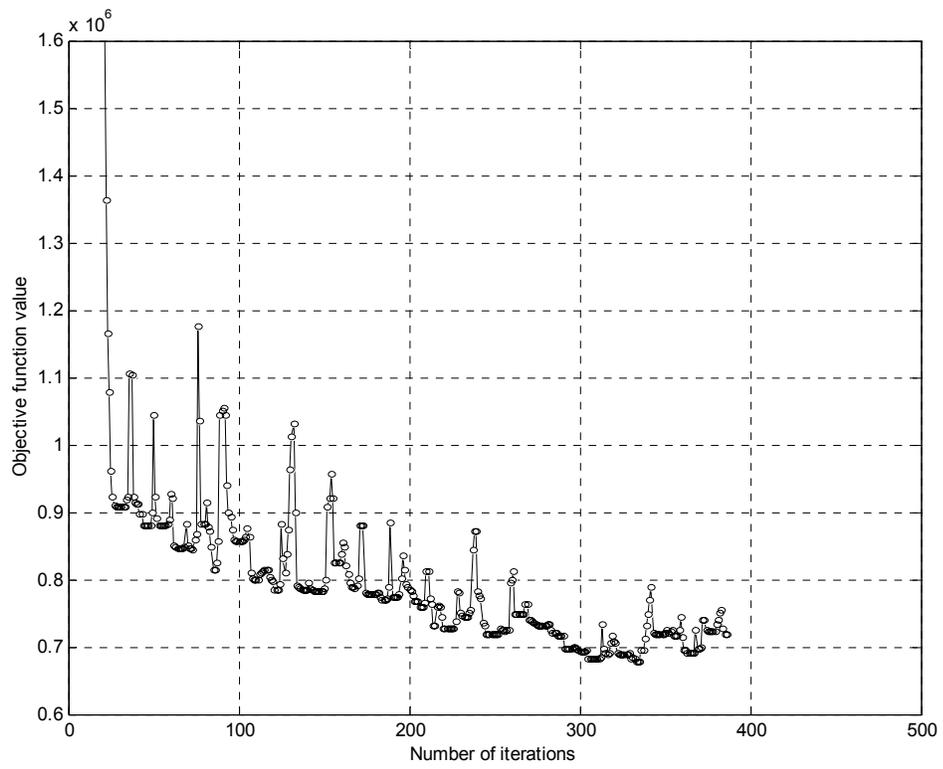
Figure 6.4 Search itineraries for the Monticello evacuation network optimization

Figure 6.4 (Continued)

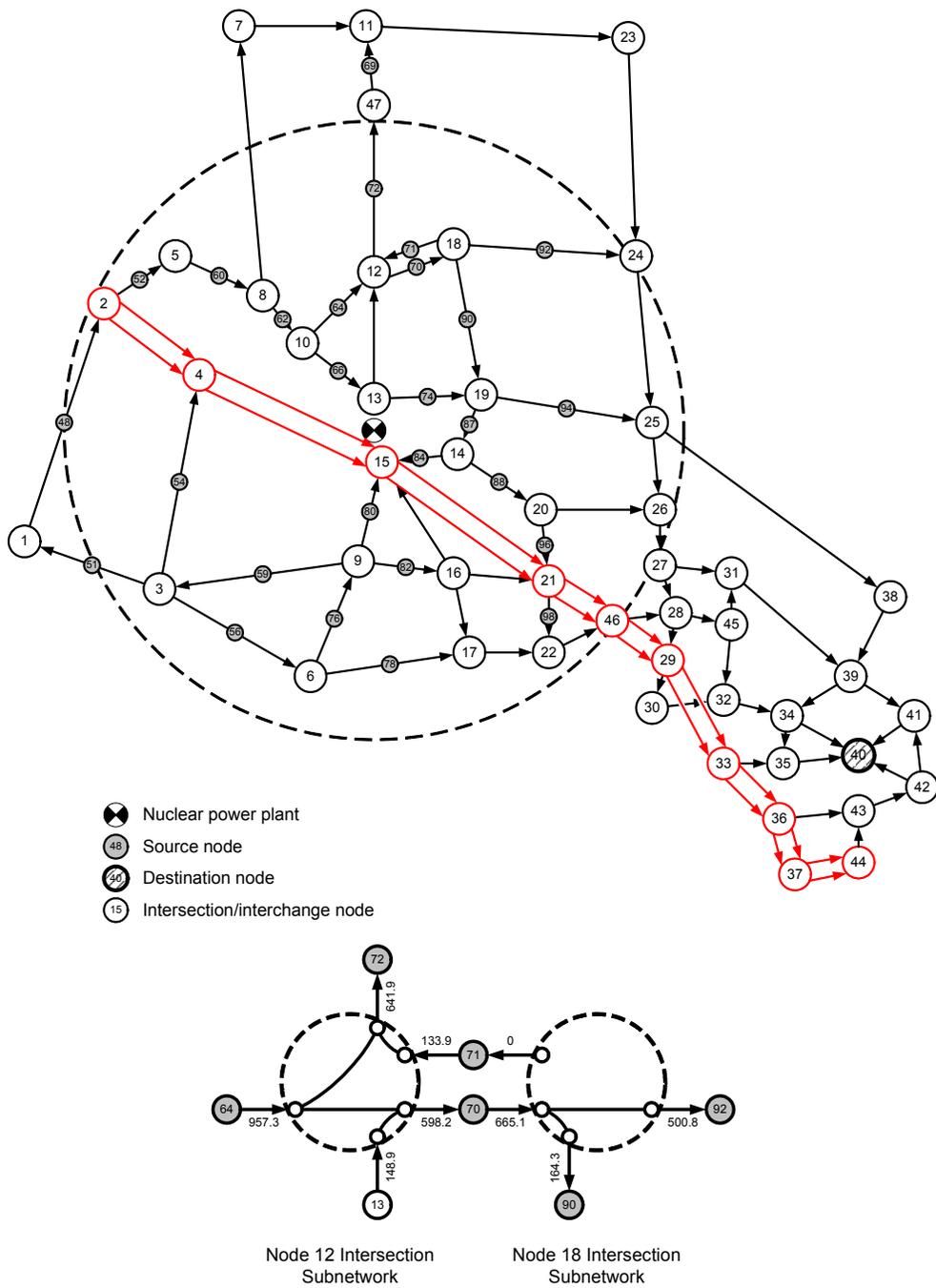


(b) Search itinerary for the Monticello network with three egresses

Figure 6.4 (Continued)



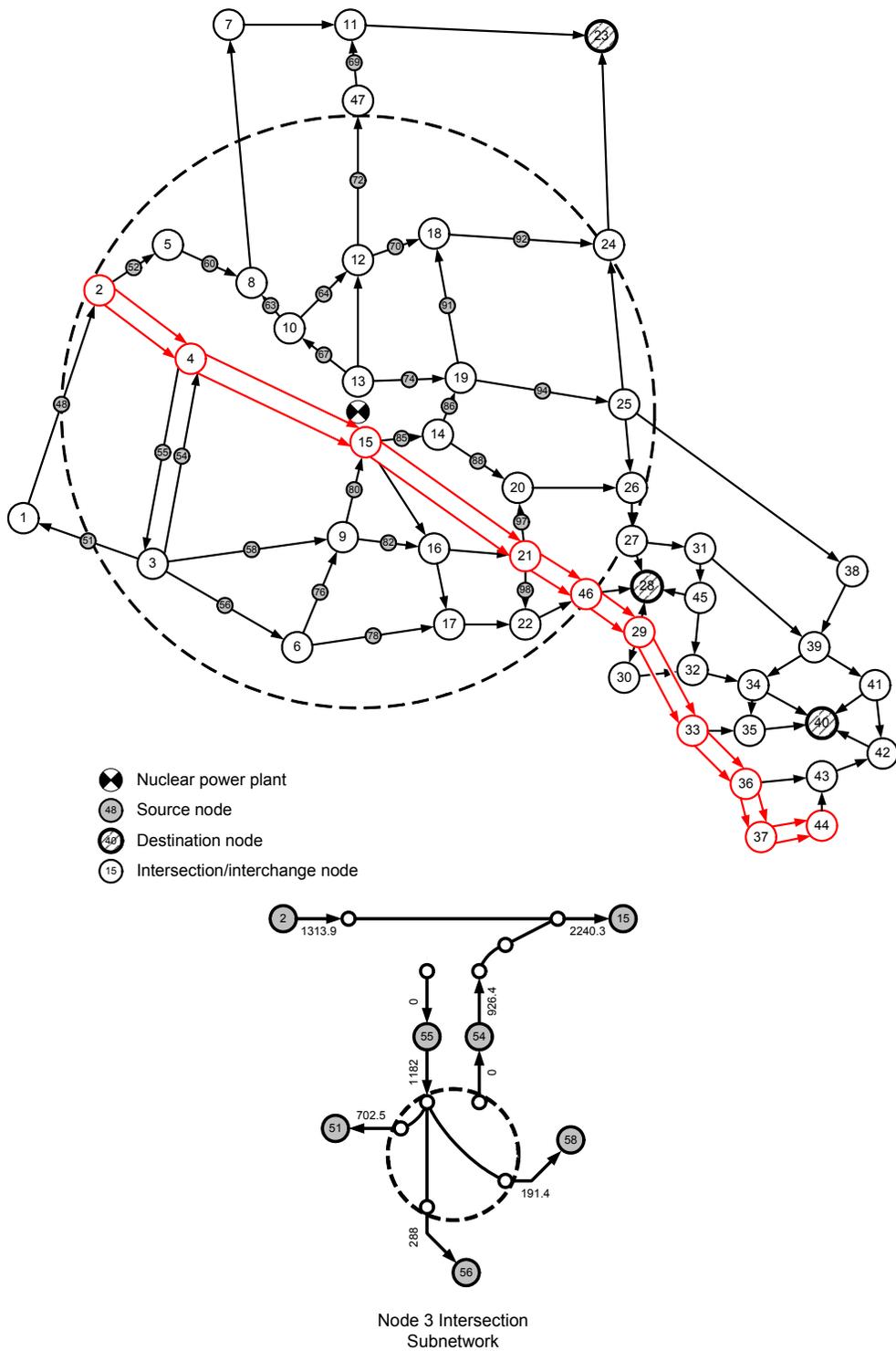
(c) Search itinerary for the Monticello network with four egresses



(a) The optimized solution of the Monticello network with one egress

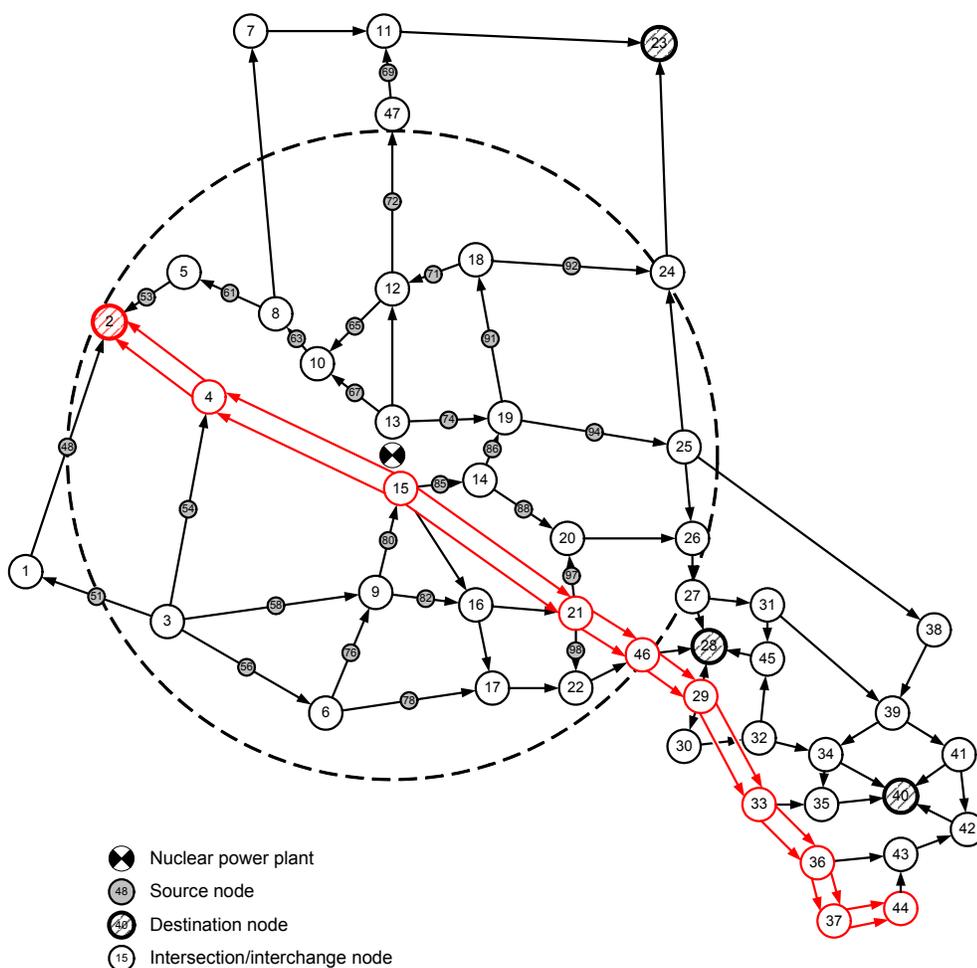
Figure 6.5 Optimized solutions of the Monticello evacuation network

Figure 6.5 (Continued)



(b) The optimized solution of the Monticello network with three egresses

Figure 6.5 (Continued)



(c) The optimized solution of the Monticello network with four egresses

The traffic movements at an intersection do not cause any crossing point if all the intersection approaches are one-way roadway segments. However, if there are two-way approaching segments, the traffic crossing pattern at the intersection turns relatively complicated. The intersection crossing-elimination requirement has been our concern in validating any optimized solution since this feasibility requirement is only satisfied by the search process in a heuristic manner. In the optimized solutions

of each of the first two network scenarios (with one and three egresses), there exists one two-way roadway segment. If we zoom in to the adjacent intersection subnetworks of these segments, it is clear that no crossing point has been generated by any of the relevant traffic movements at the associated intersections (see [Figure 6.5](#)). Therefore, we can conclude that these heuristically optimized network solutions satisfy all the feasibility requirements.

It has been assumed that individual evacuees make their route and destination choices in a stochastic user-optimal manner. Under our modeling settings, the optimal solution is the network configuration that can accommodate the stochastic user-optimal traffic flow pattern at the minimum congestion level. Therefore, the optimal network topology can be used to describe the evacuation route and destination choices, at least approximately. The combination of lane-reversal and crossing-elimination configurations as well as the associated traffic flow rates along major highways and arterials in the optimized network depicts the spatial movements of evacuating traffic and can be used to help evacuation planners in detailing an evacuation plan.

The destination-oriented solution topology is a result of the interactions between the network capacity supply and evacuation flow demand. The egress availability is one of the most important factors affecting the traffic distribution and assignment. We have observed the difference of lane-reversal directions of many links among the three optimized network solutions, which heavily depends on the relative attractiveness of available egresses. For example, many links in the vicinity of node 40 are assigned with the same directions over the three network scenarios, since the reception center at node 40 is attractive to the traffic nearby in all the cases; however, a number of links adjacent to node 2, such as $2 \leftrightarrow 4 \leftrightarrow 15$ and $2 \leftrightarrow 5 \leftrightarrow 8 \leftrightarrow 10$, have different directions in

the optimized solution of the first network scenario (with one egress) from the second and third network scenarios (with three and four egresses), because in the former scenario node 2 is a comparatively more attractive egress to the traffic generated surrounding these links than other egresses while in the latter this egress is not available.

Although most links in these optimized network solutions are assigned with a full lane-reversal direction, the understanding and interpretation of the lane-reversal directions may not be straightforward. It can be observed that in each of the network solutions a number of links may have been assigned with a counterintuitive lane-reversal direction, or some links whose lane-reversal directions may not be necessarily consistent with our intuition. To give a few examples, see links 14→15 and 16→15 in the solution of the first scenario and link 3→1 in the solutions of the first and second scenarios. It should be realized that such a solution complexity is a result of many combinatorial supply and demand factors, which may not be readily accessible through a simple, intuitive approach. The optimization results from these network scenarios show the combinatorial complexity of the network optimization problems and justify the necessity of developing a sophisticated optimization procedure such as the LR-TS method used here.

Some other insights about the model development and solution characteristics may be derived from the above result analysis. First, it is suggested that an evacuation network optimization model with the full lane-reversal requirement might be a good approximate to solve the evacuation network optimization problem. If such an approximation is acceptable, it brings two modeling and computing advantages: 1) the number of decision variables and the number of solution spaces can be significantly

reduced; 2) the intersection crossing-elimination constraint can be automatically satisfied, which further simplifies the model structure and solution procedure. Second, the topological pattern of the resulting solutions implies in some sense the underlying spatial characteristic of a desirable evacuation network, which should be the one that can sufficiently utilize the network capacity and disperse the traffic over the whole network in a distributed manner.

6.3 Comparative evaluations

To further quantify the solution method's behavior and performance, we measured and compared below the impact of a variety of evacuation situations on the solution quality and efficiency. These different example evacuation schemes have been set on both the supply side and the demand side, such as the availability of egresses and the level of demands. In addition, we compare the solution quality of our LR-TS metaheuristic with other applicable algorithmic procedures in the literature.

6.3.1 Egress availability

The multiple network scenarios with the varying number and distribution of reception centers and network exits provide us with an opportunity in measuring the influence of the accessibility of evacuation egresses on the network performance. The three scenarios with different egress settings as well as their network solutions have been described above. We attempt to provide a more detailed scenario analysis below.

The optimized network topologies in [Figure 6.5](#) clearly show the difference of the reversed lane directions and the resulting evacuation routes in the three scenarios. In

all the optimized solutions, the Interstate highway I-94 is fully reversed to provide maximum evacuation capacity; however, due to the availability of the egress at node 2, the solution of the third scenario assigns the I-94 segment between nodes 2 and 15 with a westbound direction, which is different from the plans generated from the first and second scenarios. The lane-reversal planning part for this I-94 segment not only reduces the evacuation time by taking advantage of the egress at node 2, but also provides a routing direction for traffic traversing the I-94 segment. This result is also consistent with people's common safety-seeking sense and perception. In this case, due to a focused concern and high sensitivity to the accident, most of evacuees tend to choose an evacuation route escaping away from the accident site rather than getting close to it. At this point, the solution derived from the third scenario is the most desirable among the three.

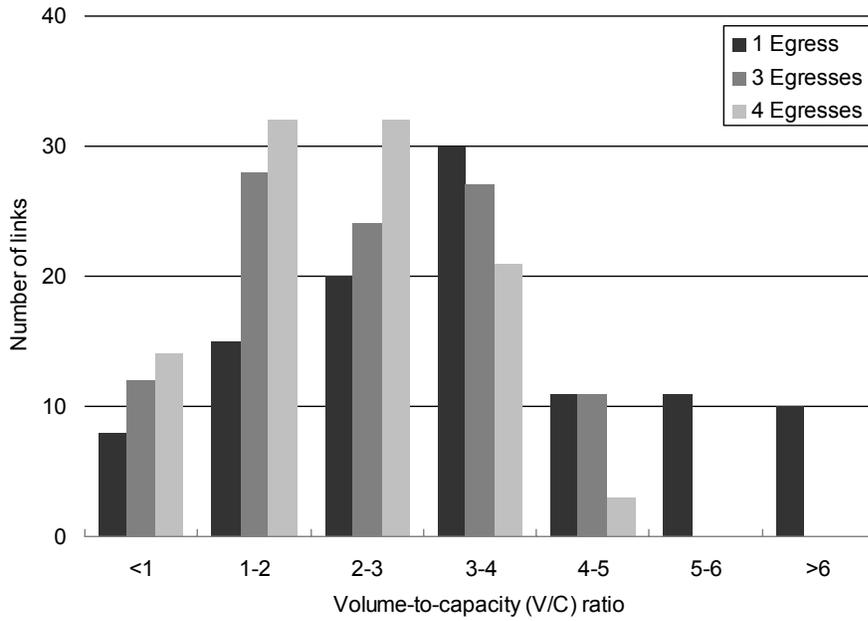
In comparing the first and second scenarios, it can be seen that the two optimized solution topologies are quite similar. However, an arterial segment between nodes 23 and 25, i.e., $23 \leftrightarrow 24 \leftrightarrow 25$, is assigned with different lane-reversal directions between the two solutions. The major reason is that the egress at node 23 is available in the second scenario and it may attract a large amount of traffic from the network. Similarly, the directions of link $31 \leftrightarrow 45$ in the two solutions are also fully reverse to each other due to the availability of the egress at node 28 in the second scenario.

More importantly, the addition of egresses not only changes the optimal solution topology, but also greatly reduces the congestion level and the evacuation time by dispersing the evacuation flow to different destinations. To quantify the benefits from adding egresses, a comparative study is conducted in terms of the network flow

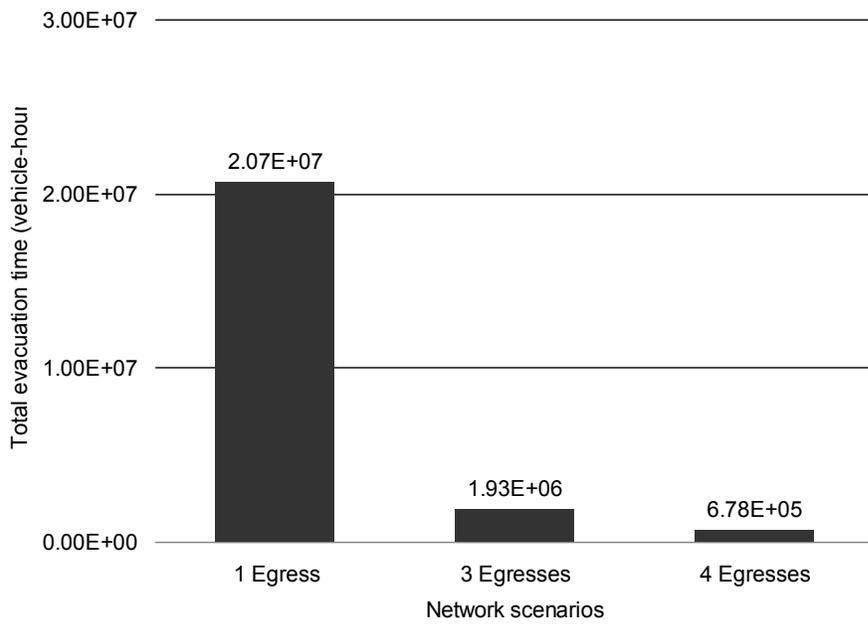
attributes such as, the volume-over-capacity (V/C) distribution, the total evacuation time, and the arrival flow split over the destinations.

The network congestion level may be assessed by the network V/C distribution and the total evacuation time. First, it is shown in [Figure 6.6\(a\)](#) that all the network scenarios encounter a congested traffic condition, in that on a large number of links is the traffic demand rate several times higher than the capacity. Not surprisingly, the optimized solution of the first scenario contains a significantly larger number of extremely congested links than the other two due to its less egress availability. While the third network scenario has only one more egress than the second scenario, it still gains a significant reduction of the network congestion. Specifically, in the third scenario, the addition of the egress at node 2 may attract many evacuees from the west part of the network rather than assign them to travel through the central and east parts of the network that has been suffered from the serious traffic congestion.

The arrival flow split over the destinations is a result of the integrated traffic distribution and assignment principle and reflects the relative attractiveness of the destinations to the evacuating population. As shown in [Figure 6.6\(c\)](#), the two scenarios with multiple reception centers (i.e., the second and third scenarios) distribute the evacuation demand in a disperse manner rather than the excessive use of a single reception center in the first scenario. Furthermore, with the addition of node 2 as an egress, the third scenario can lower and level the arrival flow rates at the other three reception centers, as compared to the second scenario. In both the cases, the reception center at node 28 attracts the most evacuating demand, due to its convenient geographic location close to the demand generation area in the network.



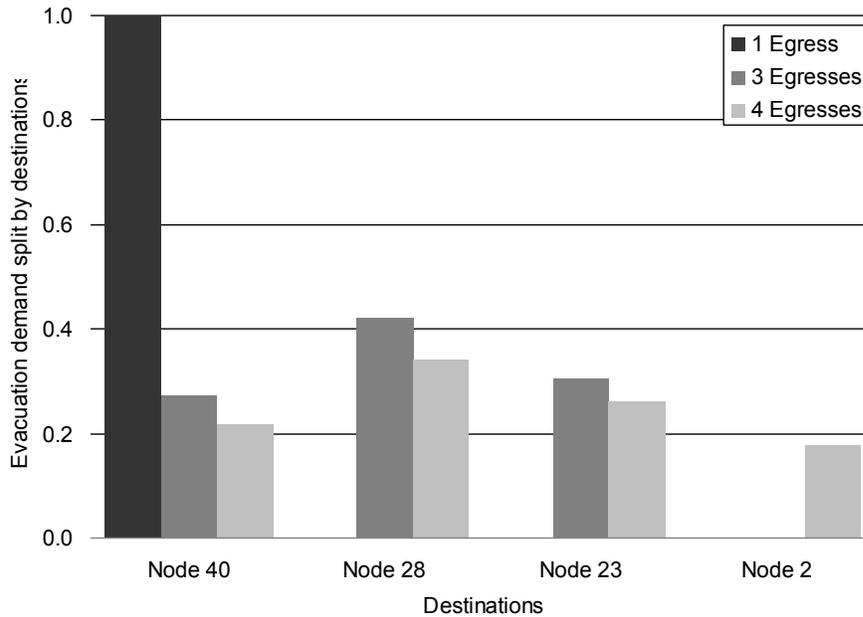
(a) The network volume-over-capacity distributions



(b) The total evacuation times

Figure 6.6 The network performance variation with the egress availability

Figure 6.6 (Continued)



(c) The arrival flow splits over destinations

The competitiveness of these egresses in the second and third scenarios may be further analyzed by a network partition analysis, by which we can identify the part of the network from which the evacuation demand is attracted by each egress. This network partition information is very useful in prescribing the evacuation plan with the lane-reversal and crossing-elimination settings. However, such a partition analysis may not be applied to describe the relative attractiveness between nodes 28 and 40, because these two nodes are both located in the southeast corner of the network. Of the two egress nodes, node 28 is much closer to the demand generation area, so an intuitive perception results in a conjecture that the importance of the reception center at node 40 would be significantly lowered if the center at node 28 is introduced into the

network because the latter center provides a nearest accommodation site to many evacuees escaping toward the east. With checking the arrival flow split, we found that in both the second and third scenarios, the reception center at node 40 still attracts a large amount of evacuating demand even if node 28 seems much more attractive in terms of the network topology. Nevertheless, the arrival flow split patterns in these multi-egress evacuation scenarios justify that the location and utilization of these reception centers in an evacuation plan provided a reasonable evacuation solution. The expected number of evacuation arrivals at each reception center can be further estimated with the arrival flow split information and thus the sufficiency of the facility capacity and relief supply at each reception center can be accordingly assessed.

From the above comparative analysis, we have seen that the third scenario with the most number of egresses is the most attractive solution among the three because it provides a most efficient evacuation network and the individual routing behavior can be best accommodated. We also want to emphasize the exceeding importance of the egress availability in evacuation planning, in that it affects the evacuation efficiency at the level of up to an order of magnitude, especially when the number of egresses is relatively limited.

6.3.2 Demand level

Evacuation demand variations also affect the optimal network solution and evacuation efficiency. In many evacuation cases, it may be quite difficult to predict the demand generation pattern and amount during the evacuation period. Thus, it is important to test the solutions under a range of possible demand levels.

A more congested network typically has slower traffic assignment convergence. In our case, due to the requirement of repeatedly evaluating the objective function, the optimization process will be significantly lengthened if a higher level of evacuation demand is loaded. Despite this computational issue, there are two important reasons to investigate the influence of the demand variations on evacuation network solutions. First, the static nature of the model obligates us to focus on optimizing an evacuation network for its surge demand rate. The robustness of an optimized network solution for a range of possible demand variations needs to be estimated to some extent. Second, given an uncertain demand generation environment, our confidence and dependency on an optimized evacuation plan could be better assessed and enhanced, if we know, at least approximately, the network performance variations due to alternative demand levels.

For each of the three evacuation scenarios, we re-optimized its lane-reversal and crossing-elimination configurations under the ± 50 percent demand levels, which we refer to as the two alternative demand levels in our experiments. A comparative evaluation with using the total evacuation time as the network performance indicator is illustrated in [Figure 6.7](#).

It is clearly shown in this figure that for all the network scenarios, a higher demand level results in a more congested network and the network congestion deteriorates at an accelerating rate with the increasing demand rate. The shape of these demand-related network performance curves may be accounted for by multiple modeling factors, of which the two most important reasons are: 1) the polynomial form of the link performance function; 2) the assumption of the simultaneous evacuation demand generation. In the former setting, the degree of the polynomial determines the

congestion-increasing magnitude in the network; the latter assumption yields a higher congestion level on links that are closer to the egresses. On the other hand, it is also observed that the congestion-increasing rate could be significantly reduced with the addition of egresses. For example, if the demand level is increased from 0.5 to 1.0, the total evacuation time of the first network scenario (with one egress) is increased by up to 31 times; in contrast, with the same demand-increasing rate, the total evacuation time of the third network scenario (with four egresses) is increased by only 9 times.

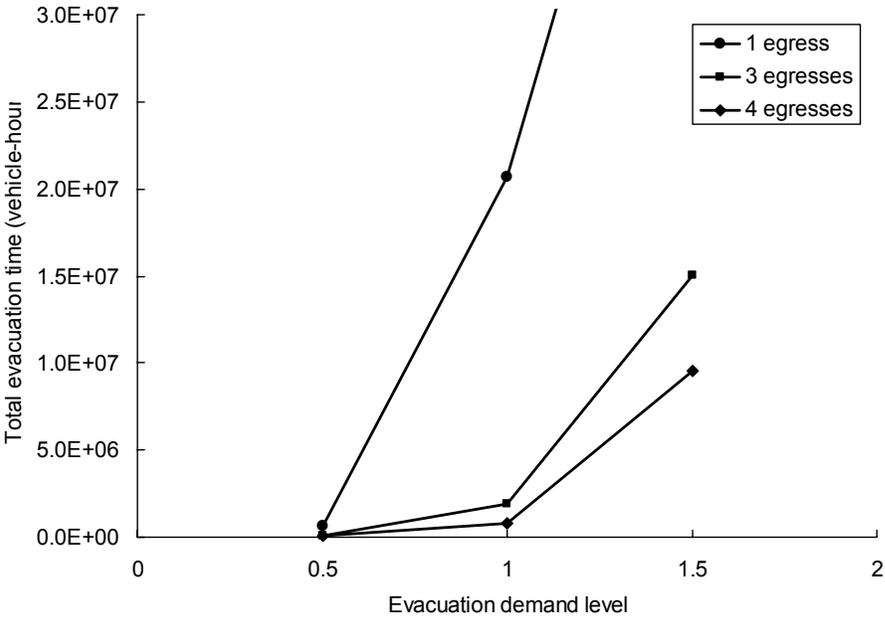


Figure 6.7 Variation of the total evacuation time over different demand levels

The network solution variation under different evacuation demand levels is another important concern in evaluating the reliability of an optimized network solution. If the network solution obtained at a slightly different demand level is significantly different from the one obtained at the expected demand level, the applicability of the model would be discounted in practice. In this case, multiple demand scenarios may need to

be developed, or a stochastic optimization approach needs to be pursued if a distribution of evacuation demand rates could be estimated. These extra efforts add the modeling complexity.

With this intention, we focus on investigating the network performance variation of the solution optimized at the expected demand level for a range of demand rates. By making use of the experiments conducted above, we applied the network solution obtained with the 1.0 demand level to the evacuation cases of the 0.5 and 1.5 demand levels and compared the resulting “sub-optimized” performance values to those of the network solutions optimized at these alternative demand levels. The discrepancy of the total evacuation time between these optimized and sub-optimized network solutions is depicted in [Figure 6.8](#).

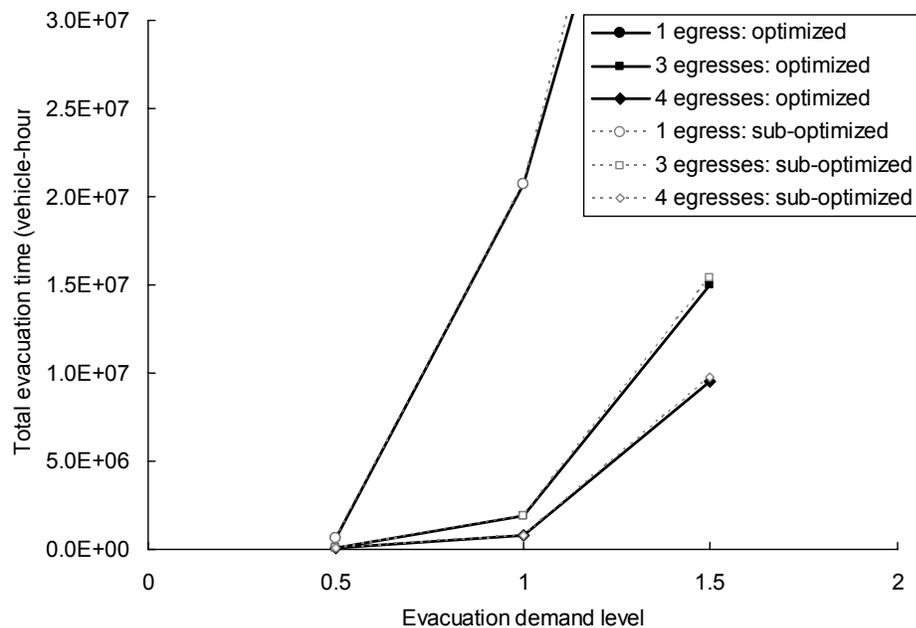


Figure 6.8 A network performance comparison between the optimized and sub-optimized network solutions at alternative demand levels

For all the three network scenarios, the comparison result shows that the network solutions optimized at the 1.0 demand level are only marginally different from the corresponding optimized solutions for the 0.5 and 1.5 demand levels in terms of the total evacuation time. The difference of the total evacuation time between the optimized and sub-optimized solutions ranges from 0.3 percent to 7.2 percent in all the cases. The average difference of the total evacuation time is merely 2.3 percent. The largest difference occurs in the case of applying the optimized solution of the first scenario to the 1.5 demand level, which is the most congested network among all the scenarios.

The comparative study described above can only be regarded as an example in its simplest case, in which the alternative demand levels are obtained by linearly increasing and decreasing the demand rates over the whole network. The demand distribution pattern, however, is not changed. A more comprehensive study should be to extend a similar comparison for optimized network solutions under a variety of possible evacuation demand patterns and levels. Nevertheless, the preliminary result obtained from this limited number of experiments justifies that an optimal evacuation network solution is capable of maintaining their near-optimal performance for a moderate range of linear-varying demand levels, at least for the Monticello evacuation network.

6.3.3 Alternative solution methods

The third comparison is focused on the aspect of solution strategies. For this purpose, two alternative algorithmic procedures are selected from the literature and applied to solve the Monticello evacuation planning problem. The first algorithm is based on a

shortest-path tree (SPT) construction procedure, as proposed by Hamza-Lup et al. (2004, 2007); the second one is a so-called flip-high-flow-edge (FHFE) method developed by Kim and Shekhar (2005, 2006), in which the lane-reversal direction on any roadway section is dependent on the congestion conditions of its two traffic directions. Although these two algorithms do not explicitly incorporate the intersection crossing-elimination requirement, the full lane-reversal assumption guarantees an automatic satisfaction of the crossing-elimination constraints.

```

algorithm SPT heuristic;

begin
    define reduced network  $(N, E)$ , super dummy node  $u$ ;
    delete all intermediate source nodes,  $N^- = N - \{s\}$ ;
    initialize  $d(j) = \infty$  for each node  $j \in N^-$ ,  $d(u) = 0$ ;
    apply Dijkstra's algorithm and label each  $j \in N^-$ ;
    for each couple of links  $j \rightarrow k$  and  $k \rightarrow j$  do
        begin
            if  $d(j) > d(k)$  then
                reverse all the lanes along direction  $k \rightarrow j$  to direction  $j \rightarrow k$ ;
            else
                reverse all the lanes along direction  $j \rightarrow k$  to direction  $k \rightarrow j$ ;
            end;
        end;
    end;

```

(a) The shortest-path tree (SPT) algorithm

Figure 6.9 The algorithmic procedures of the SPT and FHFE methods

Figure 6.9 (Continued)

algorithm FHFE heuristic;

begin

define reduced network (N, E) , traffic flow rate x_{js} , x_{sk} , capacity c_{jk} where $j \rightarrow s \rightarrow k$ is
a link pair;

conduct a stochastic traffic assignment to estimate x_{js} and x_{sk} ;

for each couple of link pairs $j \rightarrow s \rightarrow k$ and $k \rightarrow t \rightarrow j$ **do**

begin

if $(x_{js}/c_{jk})^{\beta+1} + (x_{sk}/c_{jk})^{\beta+1} > (x_{kt}/c_{kj})^{\beta+1} + (x_{tj}/c_{kj})^{\beta+1}$ **then**

reverse all the lanes along direction $k \rightarrow j$ to direction $j \rightarrow k$;

else

reverse all the lanes along direction $j \rightarrow k$ to direction $k \rightarrow j$;

end;

end;

end;

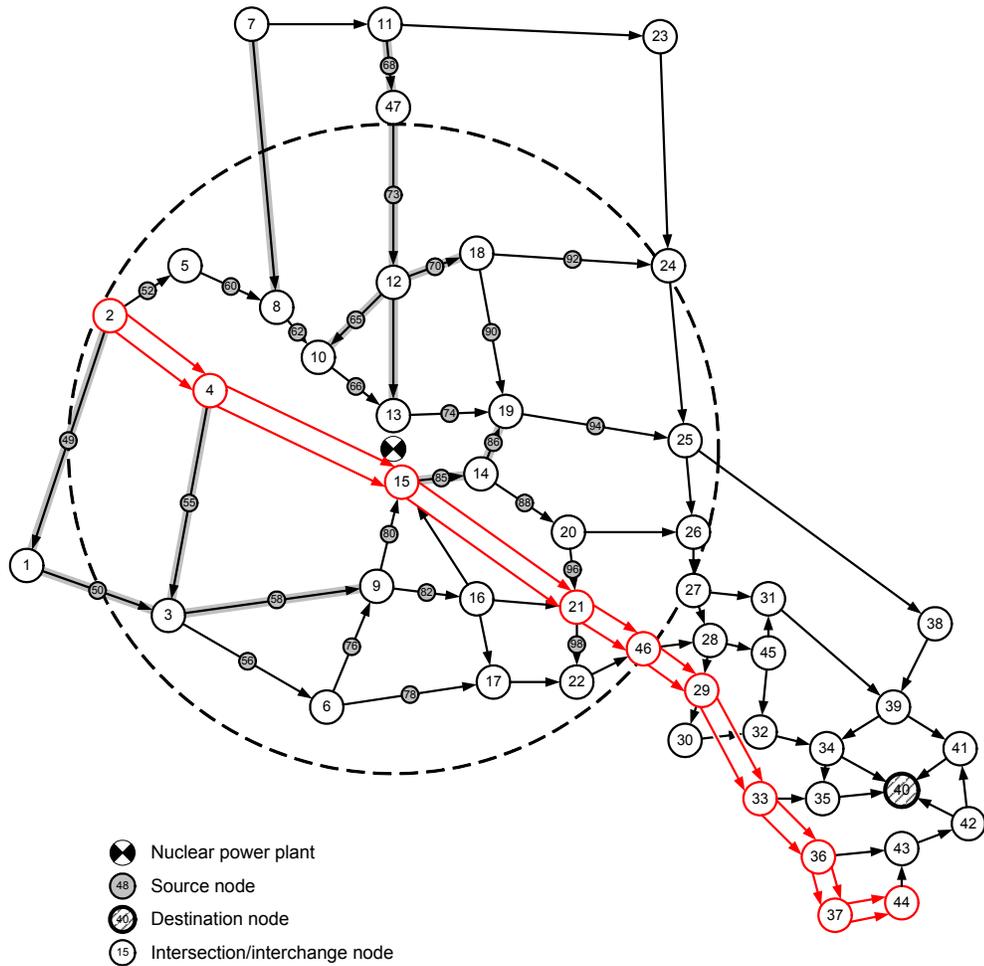
(b) The flip-high-flow-edge (FHFE) algorithm

The modified versions of the two algorithmic procedures for our specific evacuation network settings can be described as follows. In the SPT method, a shortest-path tree is first developed in terms of the static travel impedance (e.g., the free-flow travel time), starting from the super dummy destination node to all other nodes in the network, and the distance along the shortest path between any node and the destination node is labeled; the lane reversal direction of each link is then determined in terms of the distance labels of the two end nodes, that is, the direction is set from the end node with the larger distance value to the other end node with the lower value.

The FHFE method also has a two-stage process. In the first stage, a traffic assignment is carried out in the original network and the traffic flow rate on each link is recorded; the second stage resorts to a comparison of the congestion level (e.g., V/C ratio) of the two traffic directions of each roadway segment, by which the capacity of the traffic direction with the lower V/C value is fully reassigned to supplement its counter traffic direction. The pseudo-code steps of these two methods can be referred to in [Figure 6.9](#).

The prominent merit of these selected algorithmic procedures is their simple algorithmic logic and low computation cost, in which none or only one time of the objective function evaluation needs to be invoked for determining the final solution and no intersection subnetwork manipulation or optimization needs to be actually conducted. However, the optimality condition of these solutions may be subject to the following deficiencies. First, both of the methods do not explicitly consider the network optimization objective such as minimization of the total evacuation time. Second, both of the methods are not of an iterative process to monitor the traffic flow variation due to the network topology and connectivity change, in that the SPT method ignores the network congestion effect at all and the FHFE method only makes use of the congestion information at the local level and in the minimum form. Given these reasons, the two algorithmic procedures can only be regarded as heuristics for the evacuation network optimization problem defined here. Despite these algorithmic deficiencies, from a practical point of view, the simple logic and intuitive solution-deriving principle make these methods to be very attractive candidates and their solutions may be on some degree regarded as a surrogate of evacuation plans that are derived from engineering judgments.

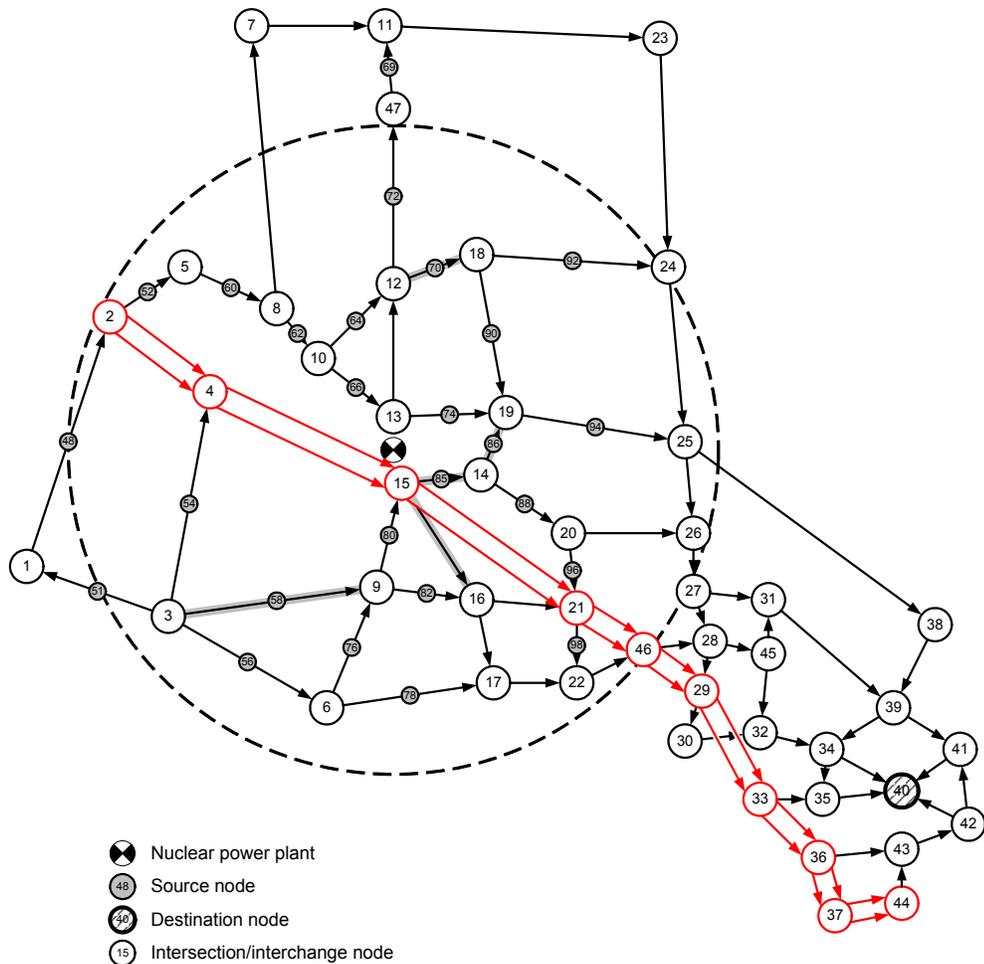
In contrast, we resorted to a relatively sophisticated solution procedure for solving the evacuation network optimization problem. A natural question may arise when we consider the relative performance between the different types of solution methods: Is a sophisticated, time-consuming metaheuristic worthwhile, compared to those simple, intuitive heuristics for the network optimization problem defined here?



(a) The network solution derived by the SPT algorithm

Figure 6.10 The Monticello network solutions from the SPT and FHFE methods

Figure 6.10 (Continued)



(b) The network solution derived by the FHFE algorithm

The solutions obtained from implementing the SPT and FHFE methods for the first scenario (with the single egress point) of the Monticello evacuation network optimization problem are presented here as an example (see Figure 6.10). It can be seen that the network-wide lane-reversal configurations in these two heuristic solutions are quite similar to the solution from the LR-TS method. The SPT and FHFE solutions contain 12 and 5 roadway sections with a different lane reversal

direction from the LR-TS solution, corresponding to 16 and 7 percent of the total number of reversible roadway sections in the network, respectively. Those different lane reversal configurations are highlighted in [Figure 6.10](#). The total evacuation time of the SPT solution is 2.92×10^7 vehicle-hours and the FHFE solution gives a total evacuation time of 2.25×10^7 vehicle-hours. Compared to the optimized LR-TS solution, these two figures are 40.6 percent and 8.7 percent higher, respectively (refer to [Figure 6.10](#)).

Two comments need to be appended here with regard to this comparison between the LR-TS metaheuristic and the two simple heuristics. First, the LR-TS method apparently outperforms the two tested simple heuristics in terms of the solution quality, at least in this Monticello network example. The FHFE method could be regarded as an attractive alternative method for the evacuation planning problem, considering its high computational efficiency in practice. However, it is expected that such a solution quality gap would be increased with the increasing the network complexity. Second, due to the structural similarity of these solutions, the solutions derived from the simple heuristics could be used as a good initial solution of our complex LR-TS method. With checking the search itinerary of the LR-TS method for the problem scenario (see [Figure 6.4\(a\)](#)), it is found that the objective function values of the SPT and FHFE solutions are comparable to that of the LR-TS solutions encountered approximately at iteration 30 and 150, respectively. If we use, for example, the FHFE solution as the starting point of our LR-TS procedure, a large number of iterations could be saved during the search process and the search procedure can focus on more important solution regions more quickly.

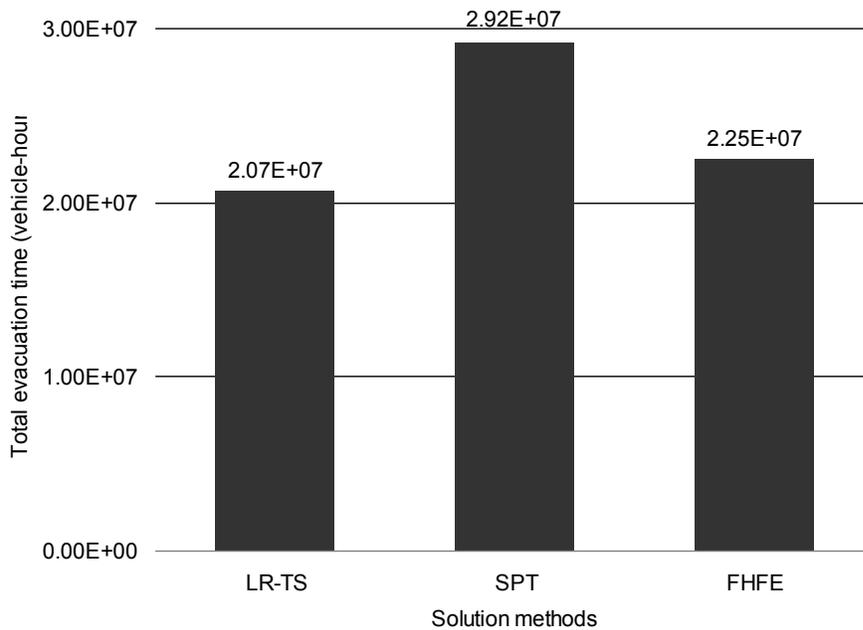


Figure 6.11 A solution quality comparison of three solution methods

6.4 Integrated evacuation network optimization and emergency vehicle assignment

Another important emergency mitigation planning component is emergency vehicle routing, allowing emergency personnel and equipment to be transported into the disaster area. This section discusses an application of the extended model for dealing with an integrated evacuation network optimization and emergency vehicle routing problem for the Monticello network. In this particular case, the primary concern of using emergency vehicle routes is to rescue casualties in case a nuclear power plant accident occurs. In accordance with this requirement, we must reserve one or more inbound routes in the evacuation network to ensure an unblocked, efficient ground transportation pipeline between the disaster area and the accessible hospitals or medical centers. Following a two-stage process based on the lexicographic optimization principle, we first analyze and tackle the emergency vehicle routing

problem and accordingly solve an evacuation network optimization problem following each emergency vehicle routing scenario; then a bi-objective scenario analysis is applied to search for the best scenario integrating evacuation network optimization and emergency vehicle routing.

6.4.1 Emergency vehicle routing

Given the locations of a medical facility and the emergency site, a single emergency vehicle route can be readily determined by the classic label-setting shortest path algorithm (e.g., Dijkstra's algorithm). Since there are a number of hospitals available for the emergency service and we must consider the influence of emergency vehicle route reservation on the evacuation network performance, the selection of emergency vehicle routes involves a two-stage procedure. The following text describes the first stage—how we developed all the candidate emergency vehicle routing schemes for the Monticello network.

A list of hospitals located in the surrounding area (including Stearns, Sherburne, Benton Morrison, Wright, Anoka and Hennepin Counties), which can provide ambulance services and treat casualties, has been identified by the Sherburne County Sheriff's Department (see [Table 6.1](#)). A regional map labeling the locations of these hospitals is shown in [Figure 6.12](#). According to the distribution of these hospital locations as well as the other emergency management requirements, three candidate routes are identified by the shortest path algorithm for the emergency vehicle use. Each of the routes serves as a transportation artery from one or more hospitals to the accident site. The selection of emergency vehicle routes depends on how many and which hospitals are needed, which is in turn related to two medical and transportation

facility attributes: the hospital capacity for the casualty treatment and the emergency route travel time between a hospital and the accident site. We are expected to provide sufficient hospital capacity to accommodate all the casualties, while to minimize the average travel time of ambulances commuting between the accident site and their affiliated hospitals. A preliminary routing analysis suggests that each of six emergency vehicle routing schemes may be used in an evacuation plan for the Monticello network:

- Route 1;
- Route 2;
- Routes 1 and 2;
- Routes 1 and 3;
- Routes 2 and 3; and
- Routes 1, 2 and 3.

All the candidate emergency vehicle routes listed in these schemes are sketched over the Monticello network in [Figure 6.13](#). Routes 1 and 2 are established on I-94, serving three hospitals in the east and seven hospitals in the west that are relatively far from the network, respectively; route 3 is on US Route 169, connecting only one hospital in the northeast corner of the network. The selection of emergency vehicle routes will be a decision-making problem subject to the emergency situation and the network traffic conditions.

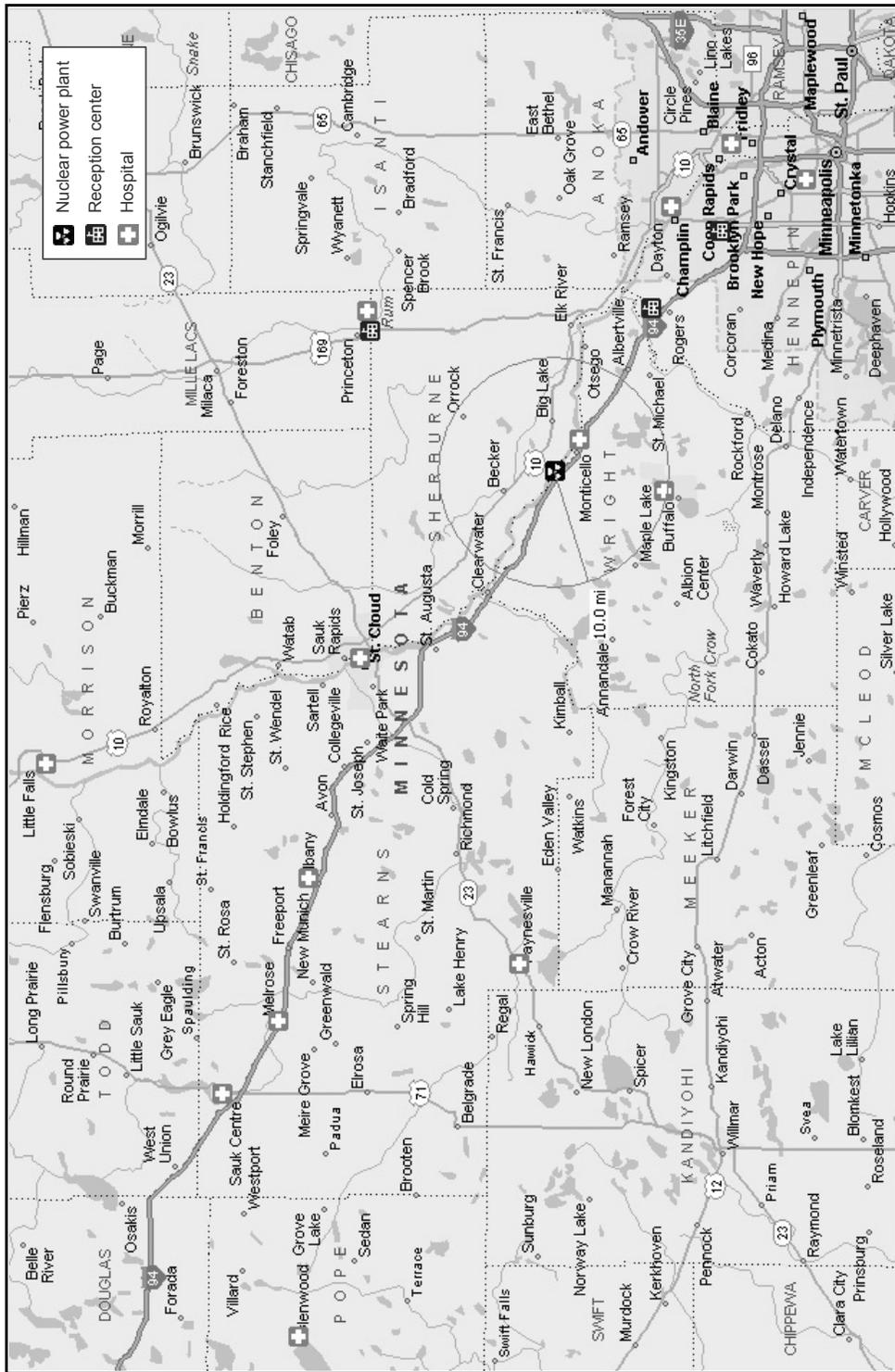


Figure 6.12 Locations of the hospitals and medical centers inside and around the emergency planning zone

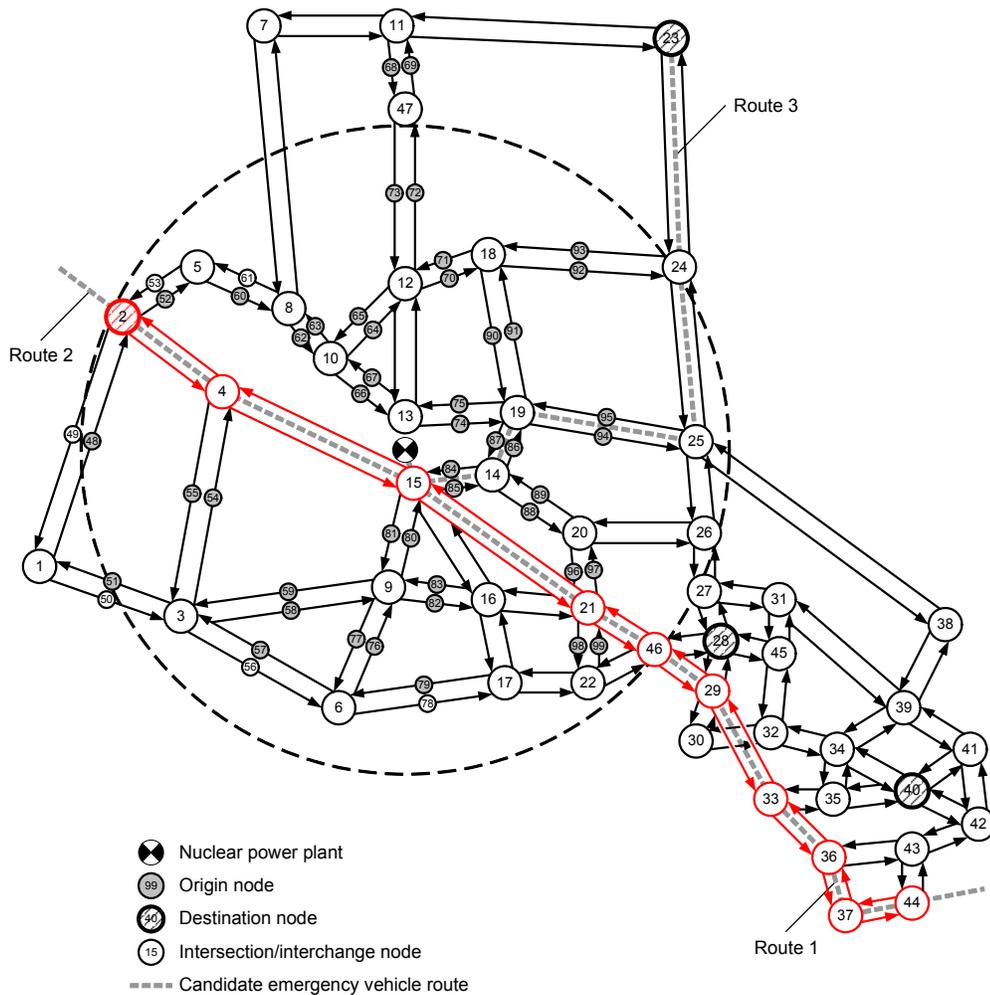


Figure 6.13 Candidate emergency vehicle routes

The introduction of emergency vehicle route planning causes a decision-making conflict with the objective of evacuation network optimization, in that the emergency vehicle routing requires reserving a certain amount of roadway capacity from the evacuation network that has been already congested, and potentially creates more traffic crossing points with the evacuating traffic at the intersections along the assigned emergency vehicle route. A simultaneous consideration of evacuation network optimization and emergency vehicle routing creates a bi-objective

optimization problem. Given the actual or estimated number and severity of disaster casualties, it is expected that a Pareto-optimal set with regard to the two objectives need to be developed so that the decision maker can determine an integrated evacuation and emergency vehicle routing plan.

Table 6.1 List of hospitals inside or around the emergency planning zone

ID	Name	Address	Accessibility	Route	Travel time
1	Saint Cloud Hospital	1406 6th Ave N Saint Cloud, MN 56303	Yes	2	35 min
2	Albany Area Hospital and Medical Center	300 3rd Ave Albany, MN 56307	Yes	2	49 min
3	St. Michael's Hospital	425 Elm St N Sauk Centre, MN 56378	Yes	2	68 min
4	Paynesville Area Hospital	200 W 1st St Paynesville, MN 56362	Yes	2	63 min
5	Melrose Hospital	11 N 5th Ave W Melrose, MN 56352	Yes	2	59 min
6	Fairview Northland Regional Hospital	911 Northland Dr Princeton, MN 55371	Yes	3	39 min
7	Monticello-Big Lake Hospital	1013 Hart Blvd Monticello, MN 55362	No	N/A	N/A
8	Mercy Hospital	4050 Coon Rapids Blvd NW Coon Rapids, MN 55433	Yes	1	38 min
9	Unity Hospital	550 Osborne Rd NE Fridley, MN 55432	Yes	1	38 min
10	North Memorial Health Care	3300 Oakdale Ave N Robbinsdale, MN 55422	Yes	1	39 min
11	St. Gabriel's Hospital	815 2nd St SE Little Falls, MN 56345	Yes	2	62 min
12	Buffalo Hospital	303 Catlin St Buffalo, MN 55313	No	N/A	N/A
13	Glacial Ridge Hospital	10 4th Ave SE Glenwood, MN 56334	Yes	2	100 min

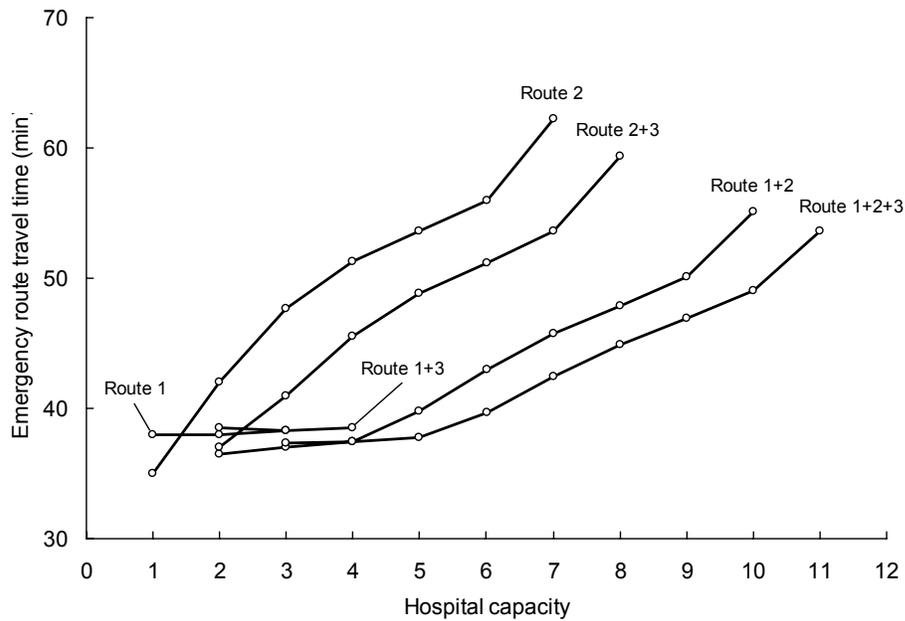


Figure 6.14 The hospital capacity and the emergency route travel time

A hospital's capacity to accommodate casualties is primarily determined by the number of its emergency rooms as well as the number of the available beds. For simplicity, we roughly believe that all hospitals on the list have a comparable number of emergency rooms and other medical facilities. Accordingly, we do not use the number of emergency rooms at each hospital to represent its capacity, but simply regard each hospital as one medical capacity unit. We calculated and depicted the shortest travel times under all the possible hospital capacity cases associated with each emergency routing scheme (see [Figure 6.14](#)).

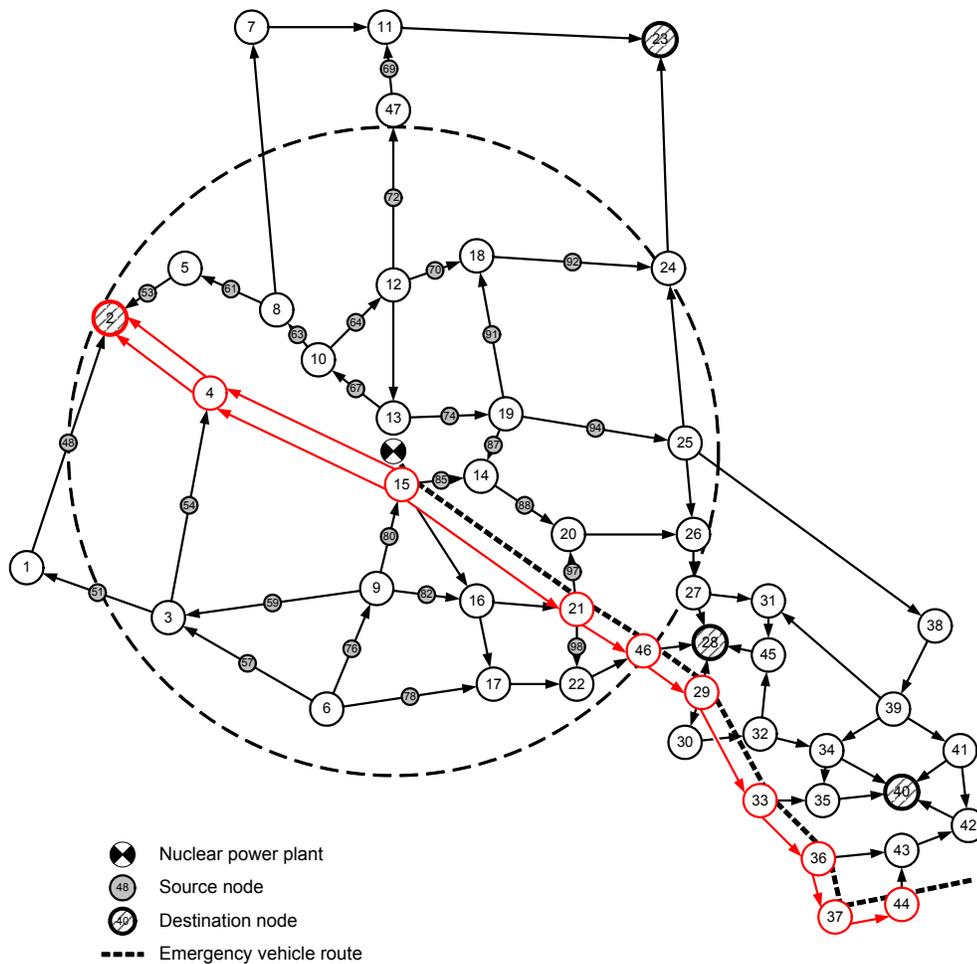
In [Figure 6.14](#), all the possible hospital capacity cases that are served by the same routing scheme are grouped together by straight lines so as to better visualize the relationship between the hospital capacity and the average route travel time under each emergency vehicle routing scheme. Please note that in this figure the average route

travel time for any hospital capacity n is the average of the route travel times from the accident site to the n closest hospitals. For example, if one hospital associated with route 1 is needed, hospital 8 or 9 (i.e., Mercy Hospital or Unity Hospital) should be used (because they are the closest hospitals) and the average route travel time is 38 min; if two hospitals associated with route 1 are required, hospitals 8 and 9 should be used together and the average travel time is still 38 min (i.e., $(38 + 38)/2 = 38$ min); if three hospitals associated route 1 are required, hospitals 8, 9 and 10 need to be used and the average route time becomes 38.3 min (i.e., $(38 + 38 + 39)/3 = 38.3$ min). Route 1 cannot provide four or more hospitals in this case. If four hospitals are required, one of the following emergency routing schemes can be used: route 2, routes 1 and 2, routes 1 and 3, routes 2 and 3, and routes 1, 2 and 3.

6.4.2 Scenario analysis of evacuation network optimization and emergency vehicle routing

At the second stage, the optimized evacuation network solutions corresponding to the six different emergency vehicle routing schemes are generated by the LR-TS search procedure, which are presented below in [Figure 6.15](#). A synthesis of the optimization results of integrating the evacuation network configuration and emergency vehicle routing are presented in [Figure 6.16](#). In this figure, we depicted a Pareto-optimal set of the bi-objective optimization problem for each hospital capacity value (i.e., from 1 hospital to 11 hospitals). We know that different routing schemes provide different numbers of available hospitals. If route 1 is used, for example, at most three hospitals can be used for the emergency rescuing service; if routes 1 and 2 are both used, we can provide 2 to 10 hospitals. It can be seen that in all the cases, the routing scheme of using routes 2 and 3 simultaneously may not be a good option, since we can always

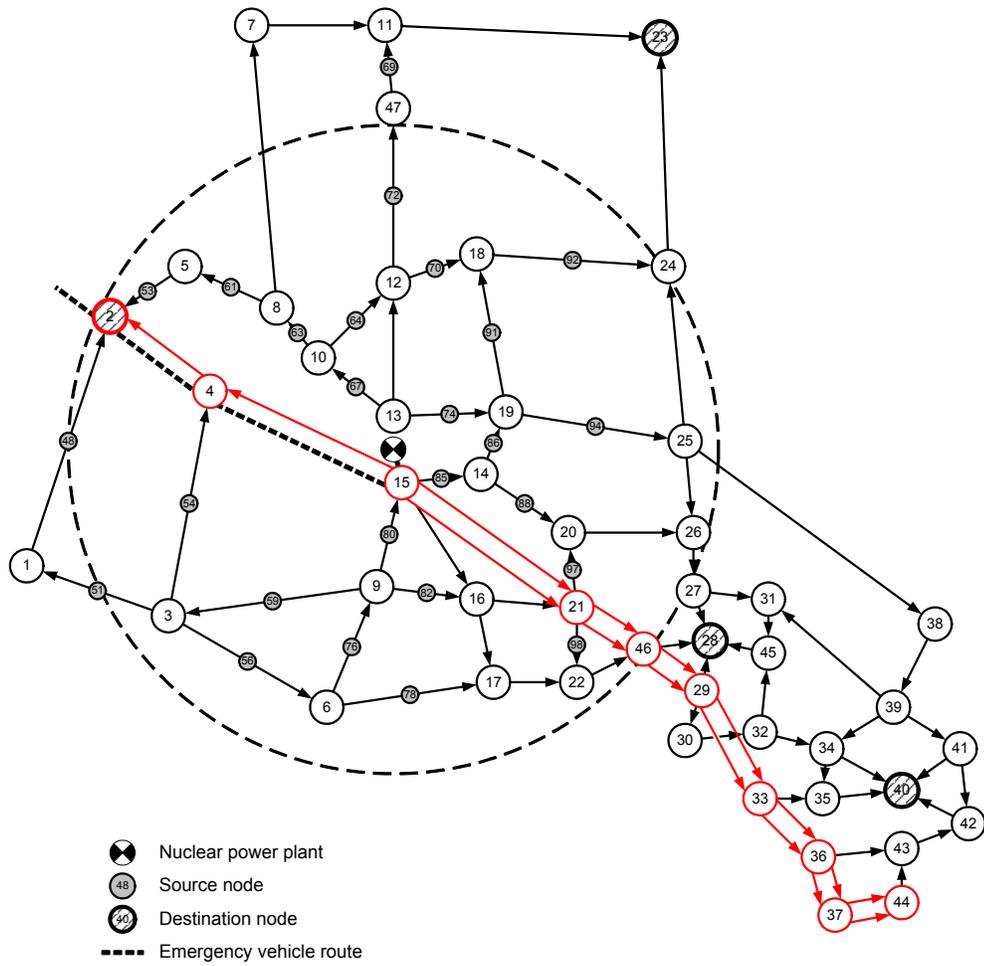
find a better routing alternative that produces both the lower network evacuation time and shorter emergency route travel time. Because of this reason, the routing scheme of routes 2 and 3 does not appear on any Pareto-optimal set in all the cases.



(a) The optimized evacuation network with emergency vehicle route 1

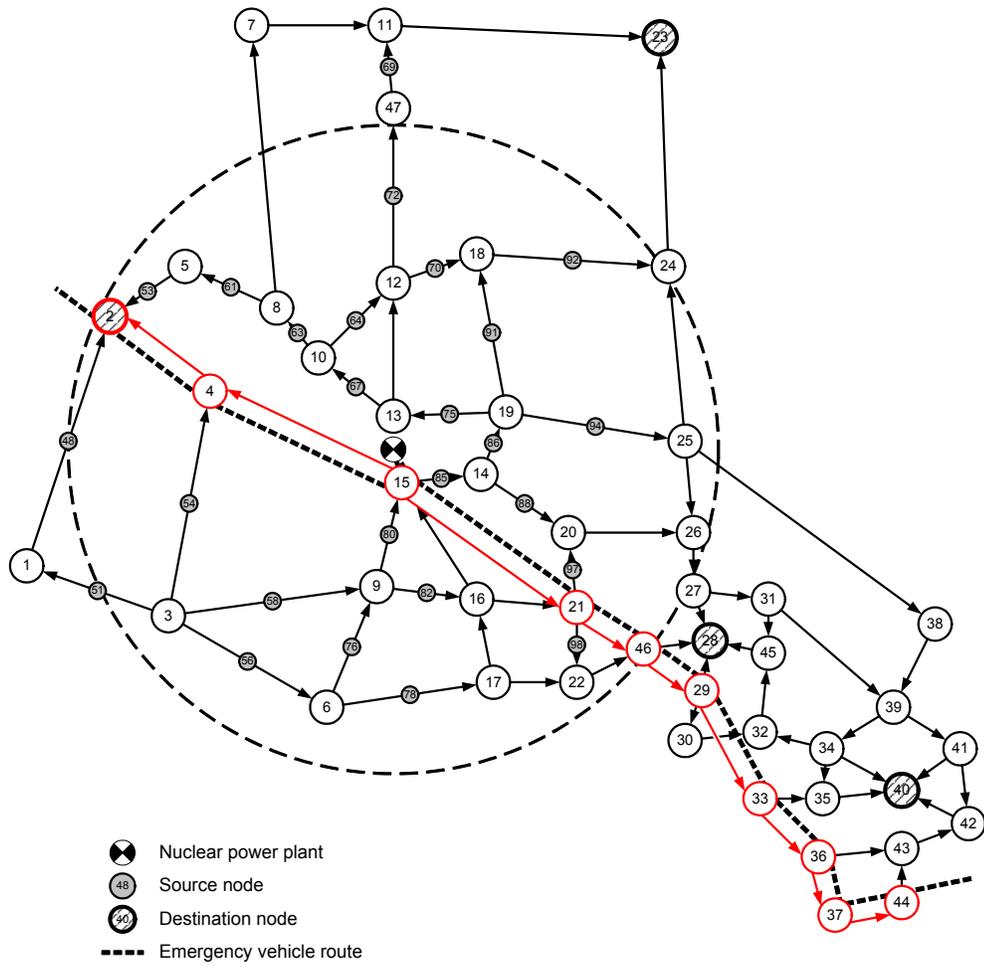
Figure 6.15 Optimized evacuation networks and emergency vehicle routes

Figure 6.15 (Continued)



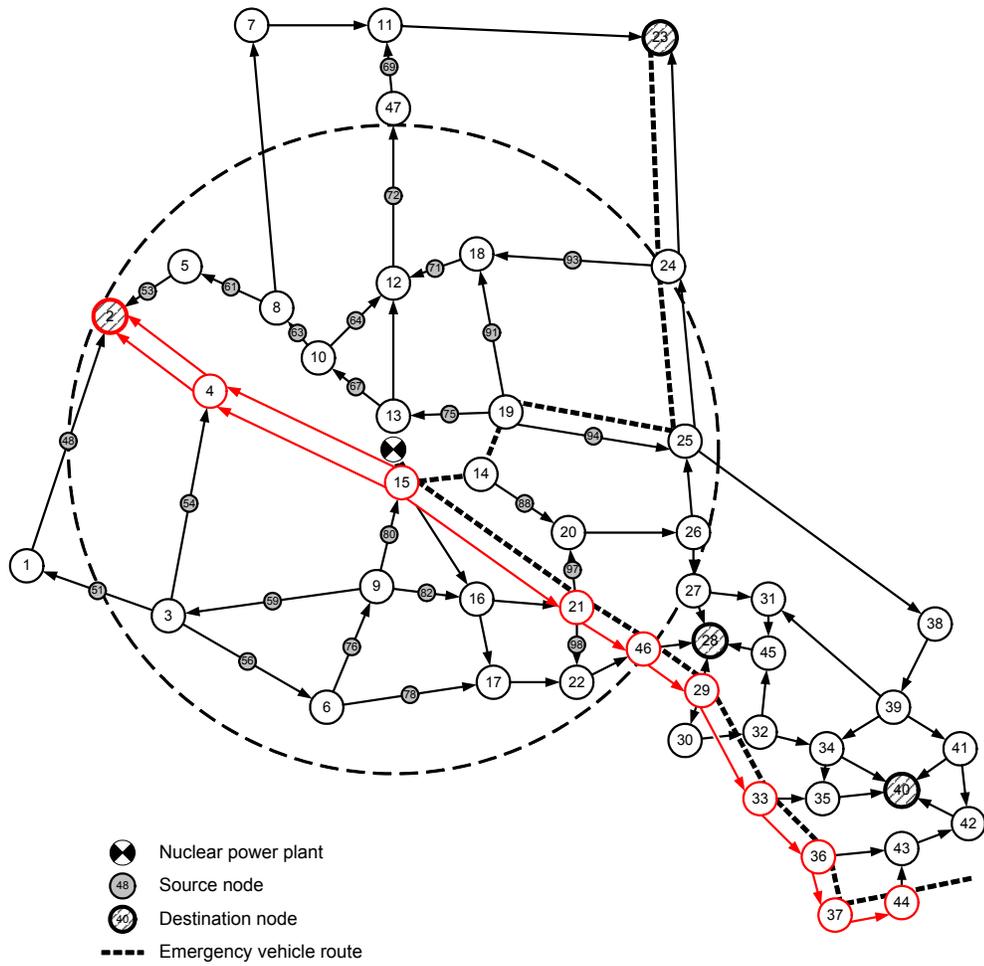
(b) The optimized evacuation network with emergency vehicle route 2

Figure 6.15 (Continued)



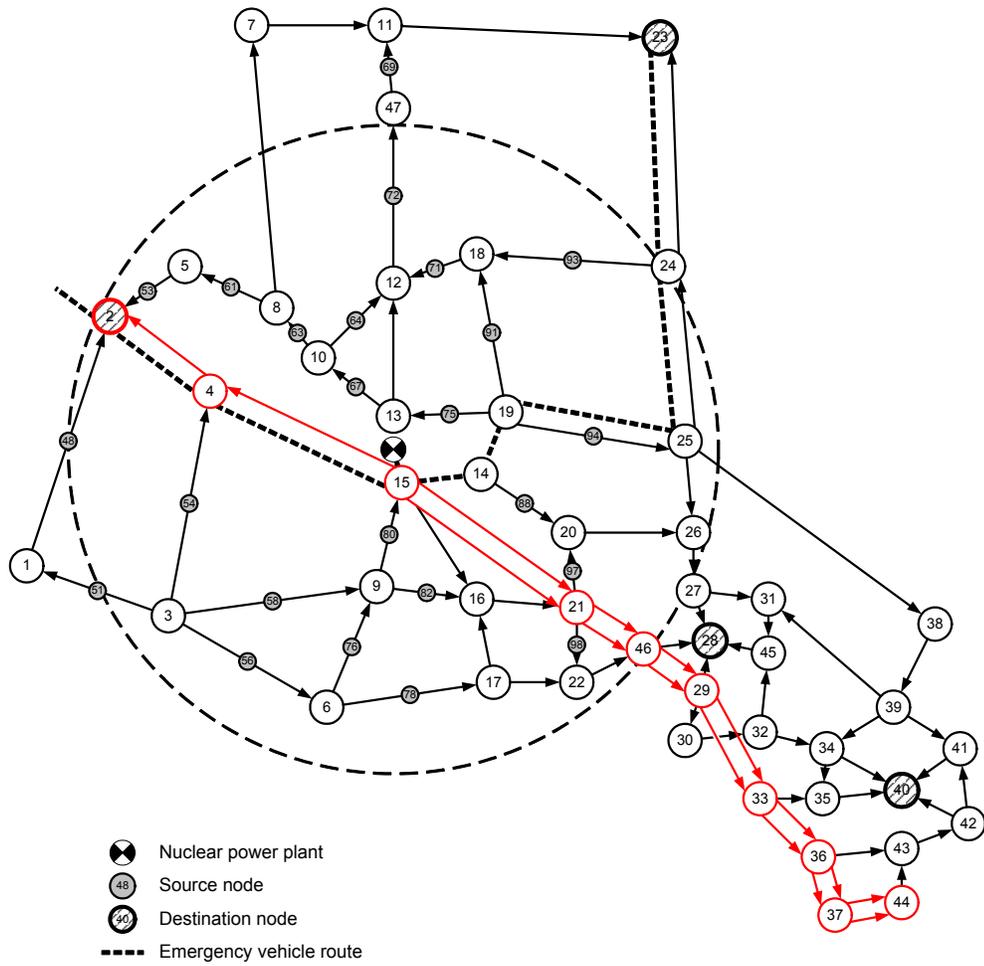
(c) The optimized evacuation network with emergency vehicle routes 1 and 2

Figure 6.15 (Continued)



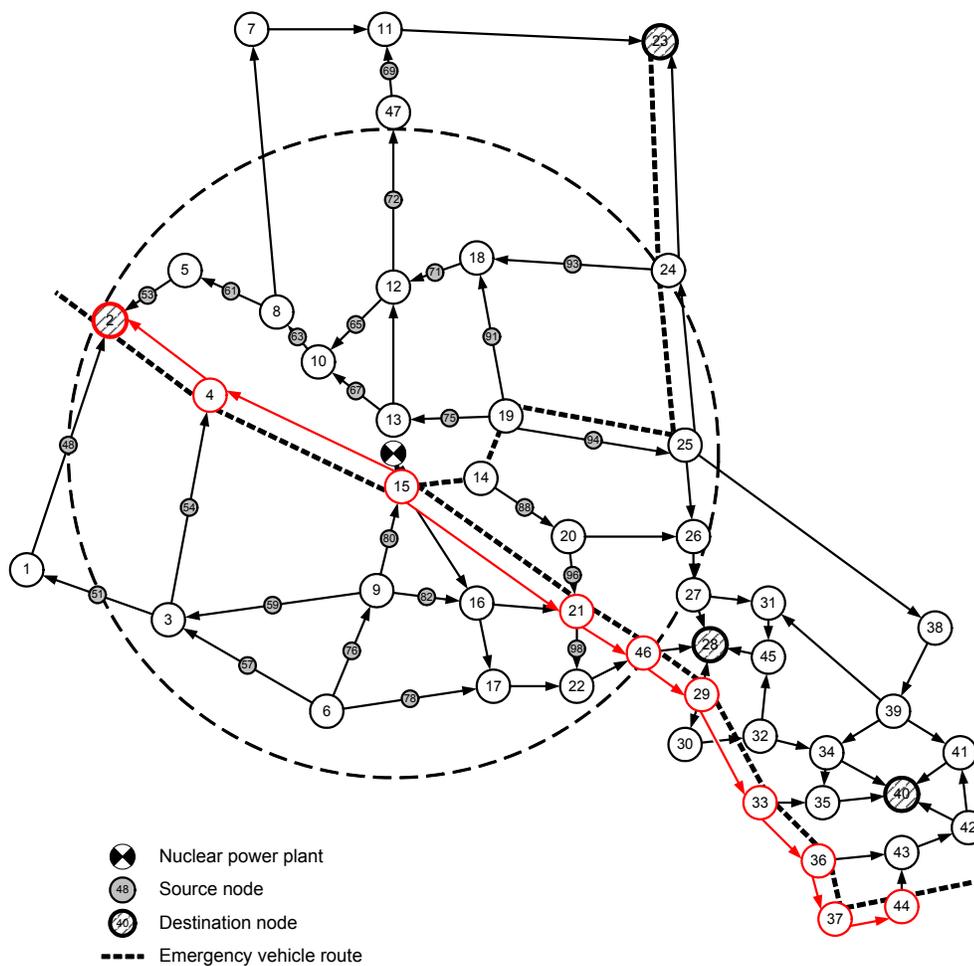
(d) The optimized evacuation network with emergency vehicle routes 1 and 3

Figure 6.15 (Continued)



(e) The optimized evacuation network with emergency vehicle routes 2 and 3

Figure 6.15 (Continued)



(f) The optimized evacuation network with emergency vehicle routes 1, 2 and 3

On the other hand, the Pareto-optimal sets indicate different scenario preferences under different hospital capacity requirements. If only one hospital is required, the network scenario with route 2 as the only emergency vehicle route is obviously the

optimal choice for both the evacuation network optimization and emergency vehicle routing. Moreover, it is readily known that the target hospital is hospital 1 (i.e., Saint Cloud Hospital). If two hospitals are required, there is no obvious optimal solution since the corresponding Pareto-optimal set includes three non-dominated scenarios, i.e., the first scenario uses route 1 with total evacuation time of 8.68×10^5 vehicle-hours and the average emergency vehicle travel time of 38 min, the second scenario uses route 2 with total evacuation time of 8.25×10^5 vehicle-hours and the average emergency vehicle travel time of 42 min, and the third scenario uses both routes 1 and 2 with total evacuation time of 1.40×10^6 vehicle-hours and the average emergency vehicle travel time of 36.5 min.

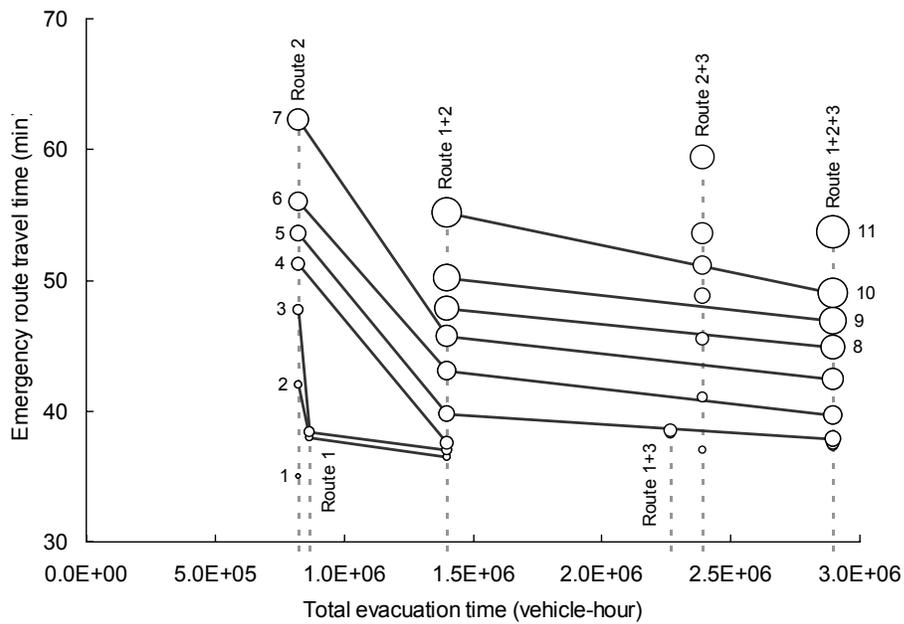


Figure 6.16 The Pareto-optimal sets of evacuation network optimization and emergency vehicle routing solutions

By considering a good trade-off between the two objectives, we might believe that the first scenario, that is, the network scenario with route 1 selected as the emergency vehicle route, is the approximately best solution, since its total evacuation time is merely slightly greater than that of the second scenario and its emergency vehicle travel time is slightly greater than that of the third scenario. When four hospitals are needed in an evacuation case, it may be difficult to determine the best network scenario since no solution can outperform others in the optimal set, even approximately. It may need to consider other factors to make a final decision in this case. As for other hospital capacity requirements, a trade-off between the different system objectives as well as the incorporation of other emergency management factors may need to be made in helping determine the optimal network solution.

CHAPTER 7

CONCLUSIONS AND FURTHER RESEARCH

*Now this is not the end.
It is not even the beginning of the end.
But it is, perhaps, the end of the beginning.
—Winston Churchill*

Evacuation planning is complex because there are many stakeholders with different perspectives, there are multiple requirements, and evacuations are nearly always surrounded by uncertainty and confusion. Past evacuation experiences have had mixed success, and there is significant need for better analytic tools to create effective evacuation plans. The focus of this dissertation is on finding the most effective ways to use existing road capacity under evacuation conditions, to minimize the total travel time of all evacuees.

To make optimal use of an existing network, we concentrate on two basic strategies—lane reversal and crossing elimination. These strategies complement one another by increasing capacity in specific directions through the network. We pose and formulate an optimization problem that seeks the set of specific link lane reversals and turn prohibitions at intersections to eliminate crossing traffic patterns, so as to minimize total travel time for evacuees.

This optimization problem is quite complex. We develop an integrated Lagrangian relaxation-tabu search (LR-TS) method to address this problem. The Lagrangian relaxation helps to decompose the problem into simpler pieces, and the tabu search heuristic is used to solve the most complex of these pieces. The computationally

intense part of the process is evaluating the objective function of the relaxed problem because that requires traffic assignment to estimate total travel time in the network and checking for traffic crossing patterns at individual intersections.

An extended model is also developed to deal with evacuation network reconfiguration when specific routes have to be reserved for emergency vehicles to access the area being evacuated (usually running counterflow to the evacuation). This problem is addressed by first identifying the candidate emergency vehicle routes and then constraining the re-configuration of the network for evacuees. There is a natural conflict between providing more direct access for emergency vehicles and providing maximum capacity for evacuees, so a series of solutions can be created as the basis for decision makers to evaluate this tradeoff.

The LR-TS metaheuristic has been tested and calibrated using a series of small test networks. These tests allowed determination of appropriate settings of parameters that control the tabu search process, in particular. Although no single set of parameter settings is likely to work best in all conditions, a likely set of values has been determined. The tests also confirm that the LR-TS method is capable of producing good solutions to the test problems with a variety of parameter values.

The calibrated algorithm has then been tested using a case study in Monticello, Minnesota. The evacuation area in that case is a prescribed 10-mile radius area around a commercial nuclear power plant. The case study confirms that the LR-TS method produces a better solution (i.e., lower total travel time) than previously available methods, but with substantially more computation. Sensitivity testing of the LR-TS solutions under varying assumptions of demand level and number of likely egress

points indicates that the solutions are quite robust, especially under demand uncertainty. This is a very important finding because estimation of the actual demand to be faced in a specific evacuation scenario is quite difficult.

The solutions in the case study illustrate some important basic properties. First, the overall performance of a solution is quite sensitive to the assumed number of egress points from the network. In many evacuation plans, there is an underlying assumption that all evacuees will go to specific designated shelter locations, but this may ignore some obvious points of egress from the network. If other egress points are recognized, the pattern of lane reversals implemented may be quite different.

Second, the solutions generally show that most links are fully reversed—that is, the link becomes one-way, rather than having only partial reversal of some lanes. The ability of the algorithm to consider partial reversals is one of the elements that contribute to its computational complexity, and there may be a useful simplification to limit the search to only full reversals. This could be implemented in a combined way also. An initial solution allowing only full reversals could be generated first, at significantly lower computational cost. This can then be used as the starting point for the existing LR-TS algorithm, to possibly refine that solution if desired.

As a third observation on the solutions, the orientation of links that are close to the egress locations is generally obvious, but we currently do not take advantage of that in the initialization of the algorithm. Thus, part of the computational effort is expended evaluating those links, when we can easily guess parts of the final solution to create a more effective starting point.

The strength of the LR-TS algorithm is in avoiding becoming stuck in local optima. However, computation is extended in moving from a “do-nothing” initial solution to reasonably good alternatives. By using some specific simple rules to create a better initial solution, we can use the LR-TS method to do more of what it is good at, and less of what we could determine in another more efficient way. The possible combination of the FHFE algorithm of Kim and Shekhar (2005, 2006) (as introduced and tested in [Chapter 6](#)) to create an initial solution, together with the LR-TS algorithm to refine that solution, may offer significant potential advantages, and is an area for further examination.

The implementation of the LR-TS search procedure could also be accelerated by using parallel computing techniques. The evaluation of candidate network solutions during an iteration and the checking of intersections for crossing violations could both benefit from parallel computing. This would be likely to speed up the entire solution process substantially.

Several other directions for further research suggest themselves. It is assumed throughout this study that the lane-reversal and crossing-elimination configurations could be implemented anywhere without regard to resource constraints. However, in practice, the implementation of these decisions requires people and equipment. People may be the most limiting resource, and may limit the number of intersections at which changes can be made and maintained throughout an evacuation. A resource-constrained version of the problem studied in this thesis is likely to be of significant interest.

The formulation studied here is static, in the sense that a constant table of originating trip rates is given, representing the demand for evacuation. The assignment of those trips to the network uses a static representation of the network condition, seeking a flow pattern that approximates stochastic user equilibrium. The introduction of the stochastic element into the flow pattern helps to diffuse traffic patterns across space, but does not directly reflect the dynamic changes that are also an intrinsic part of an evacuation event. Extending this analysis to an explicitly dynamic formulation, with queuing on network links and trip origination rates that vary over time, is another very worthwhile direction for further work. From a computational standpoint, this further complicates an already complex problem, but the dynamic changes during an evacuation are so obvious that they beg to be included. Supporting such a dynamic model with accurate dynamic data is quite another problem, however.

In general, demand estimation for evacuation planning is a problematic undertaking. The sort of sensitivity analysis conducted in the case study in [Chapter 6](#), i.e., testing the solution under demand changes of ± 50 percent, is one useful step, but a more sophisticated way of including large uncertainty about how many people and vehicles are likely to try to evacuate, over what period of time, is a very important future augmentation.

Explicit incorporation of the use of buses (either transit buses or school buses) as part of the evacuation effort is another direction of useful enhancement. If it is expected that these vehicles will make multiple trips from designated boarding areas to shelters, they must be able to move relatively quickly in both directions. This may be a severe problem during the crunch of private vehicles trying to evacuate. Explicit consideration of buses in the overall solution is likely to be useful.

The type of analysis included here—reconfiguring parts of the road network to aid motorist evacuation—is only one part of a much larger effort in planning for emergency preparedness and management. A network reconfiguration plan needs to be integrated with other elements of the emergency response plan, and responsibilities for implementing the various parts of the overall plan need to be clear. The integration of the needs for emergency vehicle movement within the evacuation plan that is included in this thesis is one piece of this larger issue, but many other pieces also need to be addressed. The work here contributes to the creation of effective evacuation plans, but many other elements are also necessary for effective implementation.

APPENDIX A

STOCHASTIC NETWORK LOADING

*As far as the laws of mathematics refer to reality, they are not certain;
and as far as they are certain, they do not refer to reality.*
—Albert Einstein

A.1 The stochastic network loading procedure

The stochastic network loading procedure includes a forward pass from an origin node to all destination nodes and a backward pass from each destination node to the origin node. The forward pass starts from the origin to gradually examine all the other nodes in the network through a merging process and a scanning process. The merging process for a node is used to determine the travel time between the origin and this node (i.e., the arrival time at this node) and the probability that traffic arriving at this node uses a merging link. The key mechanism in this procedure is to use the Clark's approximation technique to approximate the merging process. The theoretical rationale and algorithmic steps of the Clark's approximation method can be seen in Maher (1992) and Maher and Hughes (1997).

For simplicity, we present only the essential procedure of the analytical network loading method below. The details of implementing Clark's approximation for the network loading can be found in Maher (1992).

Given a node, k , and the set of its arriving links, B_k , the travel time from the origin to node k by definition is,

$$C_k = \min_{a \in A_k}(C_k^a), \quad (\text{A.1})$$

where C_k^a is the arrival time of traffic that reaches node k via arc a . The probability that traffic from the origin reaches node k via link a is defined as,

$$P_a = \Pr(C_k^a = C_k). \quad a \in B_k \quad (\text{A.2})$$

After the merging process of node k is completed, the scanning process scans its leaving links and calculates the arrival time at the downstream nodes of node k through these leaving links. Given a link, b , emanating from node k to another node, l , the arrival time at node l via link b is:

$$C_l^b = C_k + T_b, \quad b \in B_l \quad (\text{A.3})$$

where T_b is the travel time of link b and B_l is the set of arriving links of node l . The merging process expands from the origin to other nodes in the network via the network connectivity until all the destinations are completed. Then, a backward pass starts from a destination to assign the traffic demand between this destination and the origin through the network, in terms of the merging probability P_a of link a . Such a backward pass needs to be conducted for each origin-destination (O-D) pair.

As we stated, the Clark's approximation method is used to approximate the arrival time of a node in the merging process and calculate the merging probability associated with this mode. In a general network setting, however, two operational problems may arise, which yield some theoretical difficulties in implementing Clark's approximation

in stochastic network loading. The first problem is the so-called deadlock problem, emerging where there exists loops in the network. Without an external remedy method, a loop may result in that the forward process for a node on a loop cannot be completed and hence prevents the proceeding of the network loading process. The second problem is how to compute the covariance between the arrival times of a node via its arriving arcs. We discuss the approaches to solving these operational problems in the following.

A.2 Loop deadlock

Maher and Hughes (1997) suggested three approaches to tackle the deadlock problem. The first loop-breaking approach is to eliminate a loop by deleting one or more loop links with little possibility of use. The second approach is an approximate method, which is to estimate the arrival time at a loop node through a loop link by other techniques (than Clark's approximation). The third approach is to add an extra convergence check to the second approach. The convergence check at its first time is to compare the arrival time of a loop node estimated initially by another technique (for example, as used in the second approach) to its value re-estimated by Clark's approximation after a forward process for the whole loop is completed. If the two values are not consistent, an iterative process of applying Clark's approximation around the loop needs to be conducted as well as the above comparison until the convergence is realized. A detailed description to the three approaches is provided in Maher and Hughes (1997).

The first approach to eliminate loops within the implementation of Clark's approximation is easiest to use among the three. However, Maher and Hughes (1997) did not suggest how to determine which link(s) on a loop may be deleted.

In the following, we introduce a simple method of searching least possible loop links. This method was originally suggested by Dial (1971) and then modified by Sheffi (1985) for a logit-based network loading scheme.

This method starts with a shortest path search from a given origin to all other nodes in the network. The shortest path search procedure gives a label to each node i , $r(i)$, which denotes the minimum travel time from the origin to node i . Any link $i \rightarrow j$ with $r(i) > r(j)$ will be regarded as an unreasonable link and be deleted prior to the implementation of Clark's approximation.

With the above criterion, we can always find at least one link on a loop satisfying $r(i) > r(j)$. Thus, at least one link along a loop is deleted and the loop is broken.

The rationale behind the link-deletion method is to relax full-loop flow by altering the network topology. No rational individual in a network would choose a path containing a loop. However, it may not always be appropriate to delete a link $i \rightarrow j$ that satisfies the criterion $r(i) > r(j)$. In fact, we merely want to prevent full-loop flow, but do not intend to affect partial-loop flow, i.e., traffic flow that uses part of the loop. If, for example, on each individual link of a loop (but not the whole loop) there is significant flow traversing, none of the loop links could be deleted reasonably; otherwise, the resulting traffic flow pattern from the link deletion would be different significantly from the original case with the loop.

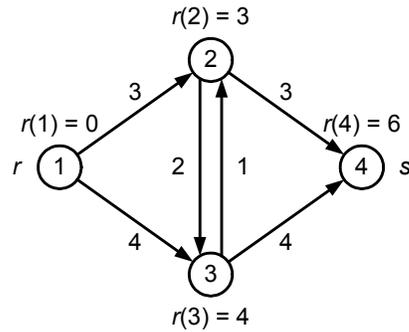


Figure A.1 An illustrative example of the loop deadlock problem

An illustrative example network for this phenomenon is given in [Figure 1](#), the topology of which was used by previous researchers, including Bell (1995), Akamatsu (1996), and Maher and Hughes (1997). In this simple network, with the given mean cost, it is readily known that link $3 \rightarrow 2$ has the property of $r(3) > r(2)$ and should be deleted according to the link-deletion criterion. The unreasonableness of deleting link $3 \rightarrow 2$ can be explained by enumerating and comparing all the feasible paths. Prior to the link deletion, we know that there may be 4 feasible paths for the stochastic network loading from origin 1 to destination 4:

Path 1: $1 \rightarrow 2 \rightarrow 4$; travel cost: 6

Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$; travel cost: 9

Path 3: $1 \rightarrow 3 \rightarrow 2 \rightarrow 4$; travel cost: 8

Path 4: $1 \rightarrow 3 \rightarrow 4$; travel cost: 8

With the deletion of link $3 \rightarrow 2$, path $1 \rightarrow 3 \rightarrow 2 \rightarrow 4$ is deleted accordingly. We may immediately find that such an outcome is intuitively inappropriate, since other two paths, $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$ and $1 \rightarrow 3 \rightarrow 4$, with greater or equal travel cost, are still kept as

reasonable paths in the network. As a consequence, if link $3 \rightarrow 2$ on the loop is deleted, the resulting traffic flow pattern in the network would be different considerably from the expected.

Nonetheless, the problem arising in the above example merely demonstrates an extreme case, which seldom emerges in real transportation networks. This seemingly theoretical difficulty within Dial's link-eliminating method should not become an overriding issue in dealing with the deadlock problem. After all, the application of Clark's approximation for probit-based network loading is inherently an approximation process.

A.3 Covariance of arrival times

In the forward process, Clark's approximation is used to estimate the overall arrival time at a node from the arrival time through each of its arriving links at this node. In this iterative approximation procedure, the covariance of the travel times through two different arriving links or through a set of arriving links and another arriving link needs to be calculated (see (4) and (9)). This in turn requires the covariance of every two arriving links of this node to be calculated. In a general network case, however, this covariance computation is not a straightforward task. The complexity is demonstrated by the following analysis.

Let us consider the covariance of the arrival times at node k via two arbitrary links, $i \rightarrow k$ and $j \rightarrow k$, i.e., $\text{cov}(C_k^{ik}, C_k^{jk})$, where i and j are the upstream nodes of these two links. First, we know $C_k^{ik} = C_i + T_{ik}$ and $C_k^{jk} = C_j + T_{jk}$ as well as $C_i \perp T_{jk}$, $C_j \perp T_{ik}$ and $T_{ik} \perp T_{jk}$, so the above covariance can be reduced to

$$\text{cov}(C_k^{ik}, C_k^{jk}) = \text{cov}(C_i, C_j) \quad (\text{A.4})$$

which is the covariance of the arrival times at nodes i and j . The value of this new covariance depends on the network overlap proportion between all the feasible paths from the origin to nodes i and those to node j . To identify the overlap proportion, a straightforward way is to enumerate all the feasible paths between the origin and node i (and j) and then make a search for the overlap proportion, which, however, is often a computationally intractable task for a network of realistic size.

With the use of Clark's approximation, a simple procedure can be used to calculate $\text{cov}(C_i, C_j)$. Given that the forward process at node j is completed earlier than node i , $\text{cov}(C_i, C_j)$ can be calculated immediately after the forward process at node i is completed. It is given as,

$$\begin{aligned} \text{cov}(C_i, C_j) &= \text{cov}(\min(\dots, C_i^l, \dots), C_j) \\ &= \sum_l p_{li} \text{cov}(C_l, C_j) \end{aligned} \quad (\text{A.5})$$

where $\text{cov}(C_l, C_j)$, $\forall l: l \rightarrow i \in B_i$, is calculated earlier by the same method. Please note that in the above recursions, $\text{cov}(C_i, C_r) = 0$, where r denotes the origin node and i is any node in the network, i.e., $\forall i \in N$. With such a recursive procedure, once the forward process of a node is finished, the covariance between this node and any other node whose merging process has been completed need to be calculated for further recursion.

We suggested this covariance computation method as an alternative to the original method suggested by Guo et al. (2001). The different point between our alternative method and the original one is that in the original method, the covariance between the arrival times of any two links are calculated and stored during the forward process, while our method instead suggests to use the covariance between the arrival times with the upstream nodes of any two links pointing to a node. Such a setting lets us merely store a covariance matrix with its size equal to the number of nodes in the network.

Apparently, the implementation of our alternative method requires less computer memory than the original method that needs to maintain a covariance matrix with the size equal to the number of links. In general, we know that the number of nodes is considerably smaller than the number of links in roadway networks. Van Vliet (1978), for example, observed that in a variety of roadway networks the ratio of links to nodes is around 3.

APPENDIX B

MINIMIZING THE NUMBER OF INTERSECTION CROSSING POINTS

*The purpose of mathematical programming is insight, not numbers.
—A.M. Geoffrion*

B.1 Problem statement

An intersection traffic crossing optimization problem is briefly defined as follows: given the inbound and outbound traffic flow rates of a four-leg intersection, the objective is to minimize the number of traffic crossing points between the traffic movements in the intersection.

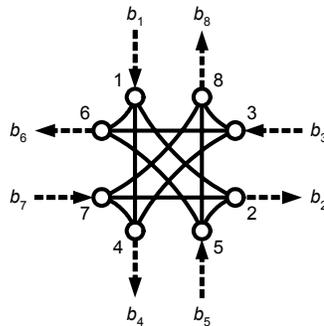


Figure B.1 The node-arc network representation of a four-leg intersection

Let us use the following example to illustrate the problem configuration. As shown in [Figure B.1](#), a typical four-leg intersection is represented by a small network with 8 nodes and 12 arcs. Each node represents either a traffic supply point or a traffic demand point. In [Figure B.1](#), nodes 1, 3, 5 and 7 are supply nodes and nodes 2, 4, 6

and 8 are demand nodes. Each arc connecting a supply node and a demand node represents a feasible traffic movement. For example, in [Figure B.1](#), arc 1→2 emanates from node 1 (supply node) to node 2 (demand node), which means a certain amount of traffic flow can be conveyed from node 1 to node 2. It is readily seen that for each supply node there are three outgoing arcs while for each demand node there are three incoming arcs.

The traffic movement tracks cross each other in the intersection. Arc 1→2, for example, which is a left-turn movement, potentially crosses arcs 3→6, 7→8, 3→4, and 5→8, if all these traffic movements are allowed. It should be noted that a right-turn movement does not cause any crossing point, e.g., arc 1→6. The objective of this intersection traffic-movement optimization problem is to find a best traffic movement configuration that minimizes the number of crossing points, subject to the traffic supply and demand requirements. With using the notation shown in [Figure B.1](#), the problem formulation can be written as:

$$\min z(y) = \sum_{ij, mn} (y_{ij} + y_{mn} - 1)^+ \quad (\text{B.1})$$

$$\text{where } (y_{ij} + y_{mn} - 1)^+ = \max(0, y_{ij} + y_{mn} - 1),$$

$$\text{s.t. } y_{ij}, y_{mn} \in \{0, 1\}, \quad \forall i \rightarrow j, m \rightarrow n \quad (\text{B.2})$$

$$x_{ij} \leq u_{ij} y_{ij}, x_{mn} \leq u_{mn} y_{mn}, \quad \forall i \rightarrow j, m \rightarrow n \quad (\text{B.3})$$

$$x_{ij}, x_{mn} \geq 0, \quad \forall i \rightarrow j, m \rightarrow n \quad (\text{B.4})$$

$$\sum_{i \in S_j} x_{ij} - b_j = 0, \text{ and} \quad \forall j \quad (\text{B.5})$$

$$\sum_{n \in R_m} x_{mn} - b_m = 0. \quad \forall m \quad (\text{B.6})$$

In this small linear integer programming model, there are two sets of decision variables, the arc variables, y_{ij} , indicating the connectivity between a supply node i and a demand node j in the intersection network, and the flow variables, x_{ij} , represents the traffic flow rate on arc $i \rightarrow j$. In the capacity constraint (i.e, constraint (B.3)), the “capacity” u_{ij} does not impose an upper bound on x_{ij} indeed, but appears merely as a sufficiently large number so as to represent the following arc-flow relationship: if $y_{ij} = 1$, $x_{ij} \geq 0$; if $y_{ij} = 0$, $x_{ij} = 0$. In the flow conservation constraints (i.e., constraints (B.5) and (B.6)), b_j and b_m are the input of the model, and S_j and T_m respectively represent the set containing the starting nodes of all the intersection arcs pointing to node j and the set containing the ending nodes of all the arcs emanating from node m , e.g., in Figure B.1, $S_2 = \{1, 7, 5\}$ and $T_3 = \{4, 6, 8\}$.

B.2 Solution algorithm

This intersection optimization problem may be solved by using the traditional branch-and-bound method due to its relative small number of search spaces. However, we consider an alternative algorithm below.

Note that the flow conservation constraints of this problem has a special structure analogous to the classic transportation problem (see Bazaraa, Jarvis and Sherali, 1990), that is, given a set of supply nodes and demand nodes, a feasible transportation flow pattern needs to be sketched between the supply and demand nodes, satisfying all the supply and demand requirements. This connection can be seen by setting nodes 1, 3, 5 and 7 as the supply nodes and nodes 2, 4, 6 and 8 as the demand nodes as well as constraint (B.5) as a demand constraint and constraint (B.6) as a supply constraint. Also, we can conveniently represent the supply and demand constraints into the so-

called transportation tableau, as shown in [Figure B.2](#), in which rows represent the supply nodes 1, 3, 5 and 7, columns represent the demand nodes 2, 4, 6 and 8, and the cell in row 1 and column 2, for example, represents flow variable x_{12} . If no flow is allowed between a supply node and a demand node, the cell in the corresponding row and column is illustrated as a shaded block. Moreover, for each supply node, the supply flow rate is indicated on the right of the corresponding row; for each demand node, the demand flow rate is indicated on the bottom of the corresponding column. The difference from the intersection optimization problem to the transportation problem is also obvious: the intersection optimization model has its extra integer requirement and its objective function is nonlinear and integer.

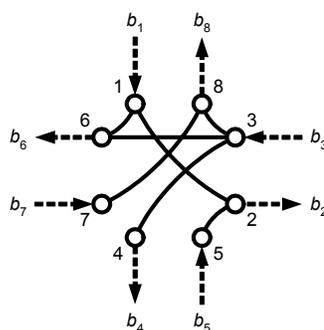
	2	4	6	8	
1	x_{12}	x_{14}	x_{16}		b_1
3		x_{34}	x_{36}	x_{38}	b_3
5	x_{52}		x_{56}	x_{58}	b_5
7	x_{72}	x_{74}		x_{78}	b_7
	b_2	b_4	b_6	b_8	

Figure B.2 The transportation tableau representation of the intersection optimization problem

It is well known that the transportation problem can be efficiently solved by the simplex method, which starts from a basic feasible solution and iteratively improve its objective function value by updating the solution from one basic feasible point to another until the optimal solution is found. A basic feasible solution of the transportation problem can be conveniently represented by a rooted spanning tree in

its transportation tableau (see Figure B.3), which contains exactly $m + n - 1$ basic variables, where m and n are respectively the numbers of supply and demand nodes.

Despite the added complexity with our defined intersection optimization problem, its structural similarity to the transportation problem inspired us to devise an efficient simplex-based iterative solution procedure, which can guarantee the optimality for the intersection optimization problem after a limited number of steps. The rationale behind this simplex-based algorithm emerges from the facts listed below.



(a) The network representation of a basic feasible solution

	2	4	6	8	
1	x_{12}		x_{16}		b_1
3		x_{34}	x_{36}	x_{38}	b_3
5	x_{52}				b_5
7				x_{78}	b_7
	b_2	b_4	b_6	b_8	

(b) The tableau representation of a basic feasible solution

Figure B.3 Representation of a basic feasible solution in the network and the tableau

For the discussion convenience, we define the following terms in describing the intersection optimization problem. Given $\mathbf{x} = (\dots, x_{ij}, \dots)$ and $\mathbf{y} = (\dots, y_{ij}, \dots)$, we call a solution (\mathbf{x}, \mathbf{y}) a basic feasible solution to the defined problem if \mathbf{x} is a basic feasible solution in the feasible region for the arc flows (i.e., constraints (B.4)-(B.6)) and \mathbf{y} is feasible. The set of all basic variables in a basic feasible solution is called the basis. Given a basic feasible solution, another basic feasible solution is called its neighbor if it can be reached by exchanging a pair of basic variables between the two solutions. All such neighboring solutions to this solution constitute its neighborhood. We also define $N(\mathbf{x})$ as the number of nonzero flow variables in solution (\mathbf{x}, \mathbf{y}) . It is obvious that $N(\mathbf{x}) \leq m + n - 1$ if (\mathbf{x}, \mathbf{y}) is a basic feasible solution of the defined problem, where $m = 4$ and $n = 4$.

Lemma 1. If a solution $(\mathbf{x}^*, \mathbf{y}^*)$ to the defined intersection optimization problem is optimal, it is a basic feasible solution; or, an alternative basic feasible optimal solution exists.

Proof. Let us assume that $(\mathbf{x}^*, \mathbf{y}^*)$ is not a basic feasible solution. By definition, this means either \mathbf{y}^* is not feasible, \mathbf{x}^* is not feasible, or \mathbf{x}^* is not basic. It is manifest that either the condition that \mathbf{x}^* or \mathbf{y}^* is not feasible contradicts the assumption given by the lemma, therefore, \mathbf{x}^* and \mathbf{y}^* must be feasible.

If \mathbf{x}^* is not basic while \mathbf{x}^* and \mathbf{y}^* are both feasible, it implies that $N(\mathbf{x}) > m + n - 1$. It reflects in the tableau that there is at least one cycle on which all the corner cells are with positive flow variables. We may adjust the flow values in these corner cells while maintaining the flow reservation feasibility until one (or more) variable, say x_{ij} ,

reaches its lower bound (i.e., $x_{ij} = 0$). The flow values in other cells of the tableau are not changed. Apparently, this procedure breaks a cycle in the tableau and produces an updated solution $(\mathbf{x}', \mathbf{y}^*)$ with fewer positive flow variables, i.e., $N(\mathbf{x}') < N(\mathbf{x}^*)$. Following this flow adjustment $\mathbf{x}^* \rightarrow \mathbf{x}'$ we can make an adjustment $\mathbf{y}^* \rightarrow \mathbf{y}'$ so as to obtain a new feasible solution $(\mathbf{x}', \mathbf{y}')$ without violating the problem feasibility by setting y_{ij} from 1 to 0 since $x_{ij} = 0$.

We can do all such adjustments until $N(\mathbf{x}) \leq m + n - 1$ and $z(\mathbf{y}')$ becomes a basic feasible solution. The immediate result from this adjustment is an improvement of the objective function value, i.e., $z(\mathbf{y}^*) \rightarrow z(\mathbf{y}')$, where $z(\mathbf{y}') \leq z(\mathbf{y}^*)$. If $z(\mathbf{y}') < z(\mathbf{y}^*)$, it contradicts the assumption in the lemma that $(\mathbf{x}^*, \mathbf{y}^*)$ is an optimal solution; if $z(\mathbf{y}') = z(\mathbf{y}^*)$, then we have $(\mathbf{x}', \mathbf{y}')$ is also optimal. Therefore, we can conclude that either $(\mathbf{x}^*, \mathbf{y}^*)$ is a basic feasible solution or $(\mathbf{x}', \mathbf{y}')$ that is basic feasible is an alternative optimal solution. ■

This conclusion provides us with a theoretical foundation to devise a method that searches for the optimal solution of the intersection optimization problem along an itinerary consisting of only its basic feasible points. The iteration between two consecutive basic feasible solutions can be realized by a pivot-move neighborhood search. To guarantee that the optimality of a basic feasible solution obtained by pivot moves, we need to investigate whether a local optimal solution to its neighborhood is globally optimal. A common way to carry out this investigation is convex analysis.

We rewrite the defined linear integer programming problem into an alternative formulation as follows:

$$\min \quad z(\mathbf{x}) = \left\{ \sum_{ij, mn} (y_{ij} + y_{mn} - 1)^+ : y_{ij}, y_{mn} \in \{0, 1\}, x_{ij} \leq u_{ij}y_{ij}, x_{mn} \leq u_{mn}y_{mn}, \forall ij, mn \right\} \quad (\text{B.7})$$

$$\text{s.t.} \quad x_{ij}, x_{mn} \geq 0, \quad \forall i \rightarrow j, m \rightarrow n \quad (\text{B.4})$$

$$\sum_{i \in S_j} x_{ij} - b_j = 0, \text{ and} \quad \forall j \quad (\text{B.5})$$

$$\sum_{n \in R_m} x_{mn} - b_m = 0. \quad \forall m \quad (\text{B.6})$$

This new problem formulation has the same structure as the transportation problem except for the objective function. It is readily known that the feasible region of this problem is a bounded polyhedral set. The remaining problem is the convexity property of the objective function $z(\mathbf{x})$. Let us consider $f(\lambda) = z(\lambda\mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2)$ and $g(\lambda) = \lambda z(\mathbf{x}_1) + (1 - \lambda)z(\mathbf{x}_2)$, given that \mathbf{x}_1 and \mathbf{x}_2 are any two feasible solutions and $0 < \lambda < 1$. It is not difficult to know that both $f(\lambda)$ and $g(\lambda)$ can be expressed as the sum of the following terms, respectively:

$$\begin{aligned} f(\lambda) &= z(\lambda\mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2) \\ &= \sum_{ij, mn} (y'_{ij} + y'_{mn} - 1)^+ \end{aligned}$$

where

$$y'_{ij}, y'_{mn} \in \{0, 1\}, \lambda x_{ij}^1 + (1 - \lambda)x_{ij}^2 \leq u_{ij}y'_{ij}, \lambda x_{mn}^1 + (1 - \lambda)x_{mn}^2 \leq u_{mn}y'_{mn},$$

and

$$\begin{aligned} g(\lambda) &= \lambda z(x_1) + (1 - \lambda)z(x_2) \\ &= \sum_{ij, mn} [\lambda (y_{ij}^1 + y_{mn}^1 - 1)^+ + (1 - \lambda)(y_{ij}^2 + y_{mn}^2 - 1)^+] \end{aligned}$$

where

$y_{ij}^1, y_{mn}^1 \in \{0, 1\}$, $x_{ij}^1 \leq u_{ij}y_{ij}^1$, $x_{mn}^1 \leq u_{mn}y_{mn}^1$, $\forall i \rightarrow j, m \rightarrow n$, and

$y_{ij}^2, y_{mn}^2 \in \{0, 1\}$, $x_{ij}^2 \leq u_{ij}y_{ij}^2$, $x_{mn}^2 \leq u_{mn}y_{mn}^2$, $\forall i \rightarrow j, m \rightarrow n$.

To compare the values of $f(\lambda)$ and $g(\lambda)$, consider the following three conditions: if given $x_{ij}^1x_{mn}^1 = 0$ (i.e., either $x_{ij}^1 = 0$ or $x_{mn}^1 = 0$) and $x_{ij}^2x_{mn}^2 = 0$, $(y'_{ij} + y'_{mn} - 1)^+ = 0$ and $\lambda(y_{ij}^1 + y_{mn}^1 - 1)^+ + (1 - \lambda)(y_{ij}^2 + y_{mn}^2 - 1)^+ = 0$; if $x_{ij}^1x_{mn}^1 > 0$ and $x_{ij}^2x_{mn}^2 = 0$, we obtain $(y'_{ij} + y'_{mn} - 1)^+ = 1$ and $\lambda(y_{ij}^1 + y_{mn}^1 - 1)^+ + (1 - \lambda)(y_{ij}^2 + y_{mn}^2 - 1)^+ = \lambda$; if $x_{ij}^1x_{mn}^1 = 0$ and $x_{ij}^2x_{mn}^2 > 0$, we obtain $(y'_{ij} + y'_{mn} - 1)^+ = 1$ and $\lambda(y_{ij}^1 + y_{mn}^1 - 1)^+ + (1 - \lambda)(y_{ij}^2 + y_{mn}^2 - 1)^+ = 1 - \lambda$; if $x_{ij}^1x_{mn}^1 > 0$ and $x_{ij}^2x_{mn}^2 > 0$, $(y'_{ij} + y'_{mn} - 1)^+ = 1$ and $\lambda(y_{ij}^1 + y_{mn}^1 - 1)^+ + (1 - \lambda)(y_{ij}^2 + y_{mn}^2 - 1)^+ = \lambda + (1 - \lambda) = 1$. Combining all these conditions, we know that $f(\lambda) \geq g(\lambda)$ holds for any $0 < \lambda < 1$. Therefore, $z(\mathbf{x})$ is a concave function[†].

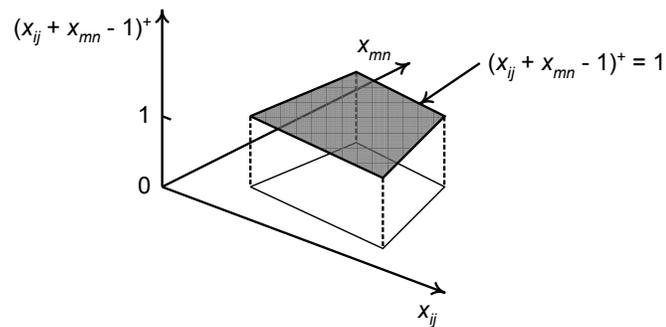
Given that the feasible region is a convex set but the objective function is a concave function, we cannot in general guarantee the global optimality of a local optimum. However, for the defined intersection optimization problem with its special structure, we can show that no local optimum can be actually held.

Lemma 2. If a basic feasible solution to the defined intersection optimization problem is a local optimal solution to its neighborhood, it is also a global optimal solution.

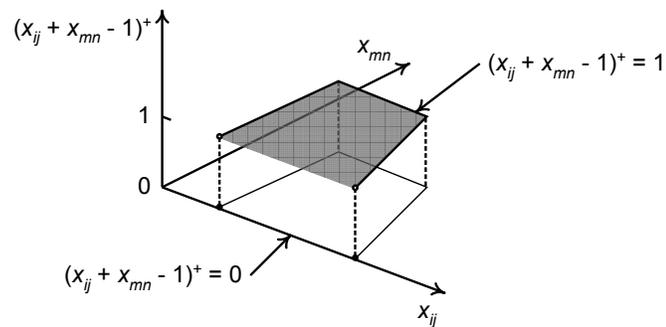
Proof. We can distinguish flow variables in two types: 1) “right-turn” flow variables, which do not impose any traffic crossing points; and 2) “left-turn” and “through” flow variables, which would potentially cause crossing points. The value of the objective

[†] Given the integer characteristic, we know that $z(\mathbf{x})$ is a stepwise concave function.

function is determined by the values of the “left-turn” and “through” flow variables. Suppose that x_{ij} and x_{mn} are two variables of the second type and their corresponding arcs may have a potential crossing point. The distribution of values of $(x_{ij} + x_{mn} - 1)^+$ is shown in Figure B.4, in which the feasible region for x_{ij} and x_{mn} are the projection of the whole feasible region of \mathbf{x} on the plane of x_{ij} and x_{mn} . Needless to say, $(x_{ij} + x_{mn} - 1)^+$ has two values: when either $x_{ij} = 0$ or $x_{mn} = 0$, $(x_{ij} + x_{mn} - 1)^+ = 0$, and when both $x_{ij} > 0$ and $x_{mn} > 0$, $(x_{ij} + x_{mn} - 1)^+ = 1$.



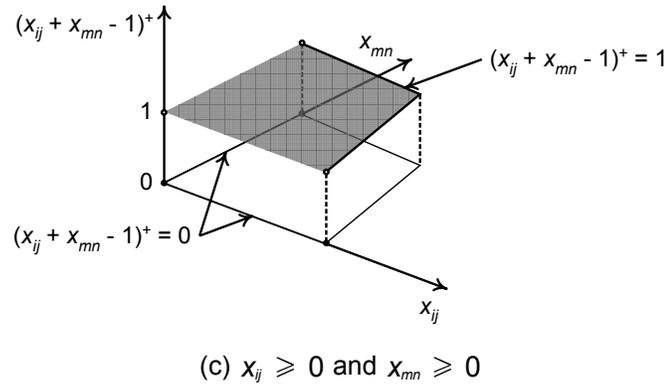
(a) $x_{ij} > 0$ and $x_{mn} > 0$



(b) $x_{ij} > 0$ and $x_{mn} \geq 0$

Figure B.4 Feasible region of a pair of flow variables x_{ij} and x_{mn} with a potential crossing point and the corresponding $(x_{ij} + x_{mn} - 1)^+$ value

Figure B.4 (Continued)



Note that for any pair of x_{ij} and x_{mn} with a potential crossing point between their arcs, its feasible region subject to constraints (B.4)-(B.6) and the corresponding value distribution of $(x_{ij} + x_{mn} - 1)^+$ can be represented by one of the conditions in Figure B.4. If a local optimal solution that is not globally optimal exists, there is at least one pair of x_{ij} and x_{mn} that there are two separate subregions both with $(x_{ij} + x_{mn} - 1)^+ = 0$ in its feasible region. However, none of the feasible regions includes such a case. Therefore, a local optimal solution will not be blocked from other optimal solutions and it is actually a global optimal solution. ■

The conclusions given above, however, do not guarantee the optimality uniqueness. Actually, it is possible to have multiple optimal solutions to an intersection optimality problem of the defined type, in which some solutions are basic feasible solutions and others are not. But we know that at least one of the optimal solutions is a basic feasible solution.

Now we have all the required theoretical elements to guarantee the correctness of the proposed algorithm. The algorithmic procedure of the resulting simplex-based pivot-move algorithm can be sketched as follows:

Step 1. Obtain a starting basic feasible solution as the current solution and compute its objective function value z^* . This can be accomplished by applying the northwest corner rule in the tableau (see Bazaraa et al., 1990);

Step 2. Conduct all the candidate pivot moves by entering each nonbasic variable into the basis and compute the updated objective function value with each candidate move. Choose the best move with the lowest objective function value z' ;

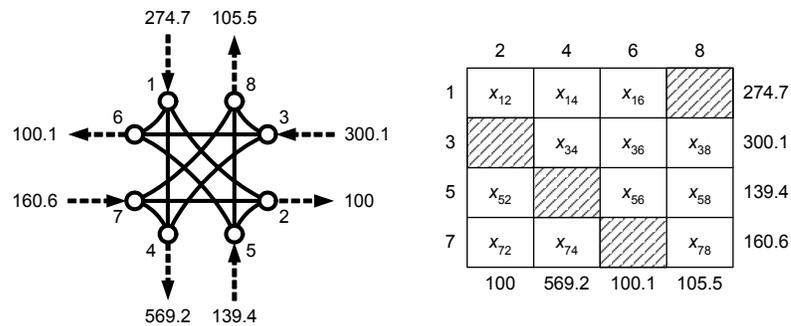
Step 3. Compare the objective function value with the best move, z' , and the current objective function value, z^* . If $z' \geq z^*$, stop the iteration and we have the optimal solution z^* at hand; if $z' < z^*$, implement the best move to obtain the updated basic feasible solution and assign $z^* = z'$, and then go to step 2.

B.3 Numerical examples

To demonstrate the validity and efficiency of the proposed simplex-based algorithm, we present a couple of numerical examples in the following.

The first example problem with its network and tableau representations is given in [Figure B.5\(a\)](#). The initial basic feasible solution derived by the northwest corner rule is shown in [Figure B.5\(b\)](#), in which the basis consists of variables x_{12} , x_{14} , x_{16} , x_{34} , x_{56} , x_{58} , and x_{74} , and the objective function value with this solution is 5. Starting from this

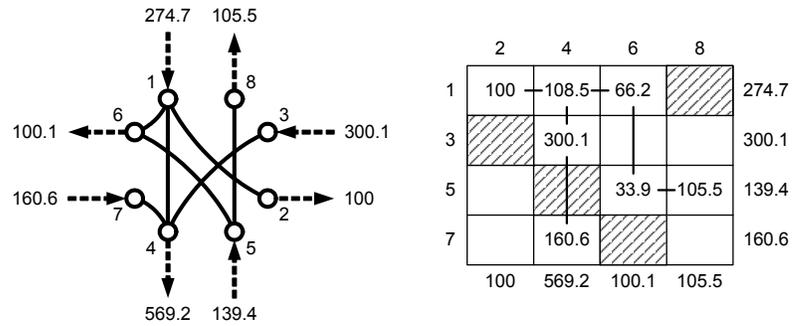
initial solution, it is found that by examining all the nonbasic variables that a pivot move that the nonbasic variable x_{52} enters the basis and the basic variable x_{36} leaves the basis yields a best move (i.e., the lowest objective function value). By implementing this move, we get an updated basic feasible solution, the basis of which includes variables x_{12} , x_{14} , x_{16} , x_{34} , x_{52} , x_{58} , and x_{74} , and the objective function value of which is 3. This updated solution is illustrated in Figure B.5(c). The same examination and pivot procedure is then applied to proceed with the search for improved solutions. Next, we obtain the basic feasible solution at iteration 2 by entering x_{38} into the basis and getting rid of x_{12} from the basis, as shown in Figure B.5(d), whose objective function value is 1. Since this solution cannot be improved by a single pivot move, we can conclude that it is the optimal solution to the problem.



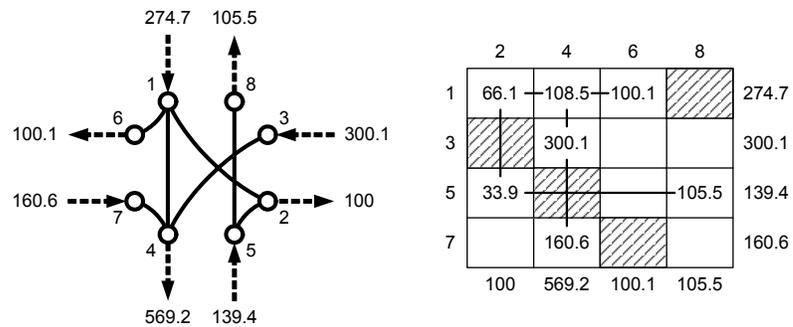
(a) The network and tableau representations of the problem

Figure B.5 The first numerical example and its solutions by the simplex-based algorithm

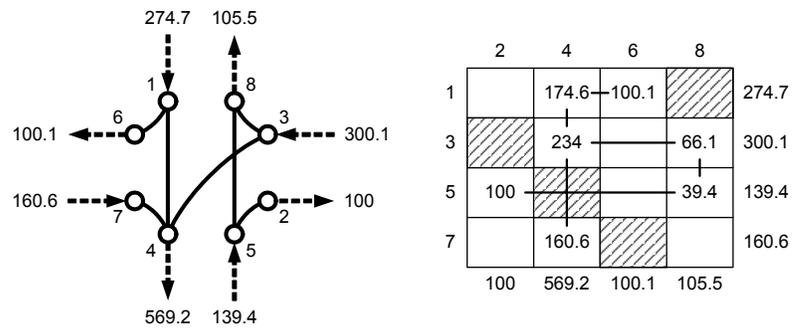
Figure B.5 (Continued)



(b) Iteration 0 (Objective function value: 5)

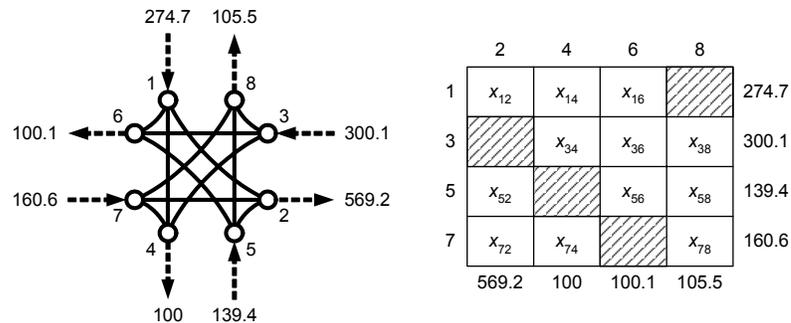


(c) Iteration 1 (Objective function value: 3)



(d) Iteration 2 (Objective function value: 1)

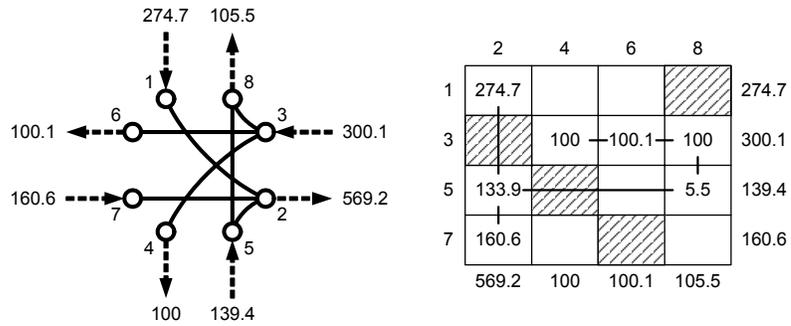
The second example is a copy of the first one except that the values of b_2 and b_4 are swapped. The initial solution obtained by applying the northwest corner rule is shown in Figure B.6(a), in which the basis consists of x_{12} , x_{34} , x_{36} , x_{38} , x_{52} , x_{58} and x_{72} and the objective function value with this solution is 7. At the first iteration, it is found that two pivot moves yields the same best objective function value (i.e., the value is 3). These two moves are respectively that x_{16} enters the basis and x_{58} leaves the basis, and x_{74} enters the basis and x_{58} leaves. Since the two pivot moves improves the objective function value by the same quantity, we can implement either of them to obtain the next basic feasible solution. For completeness, we present the basic feasible solutions resulted from both the moves respectively in Figure B.6(b) and Figure B.6(c). Further examinations on these two solutions conclude that both of the solutions are optimal to the problem since no pivot move that improves the objective function value can be found. This example demonstrates a case that more than one optimal solutions exist at the same time.



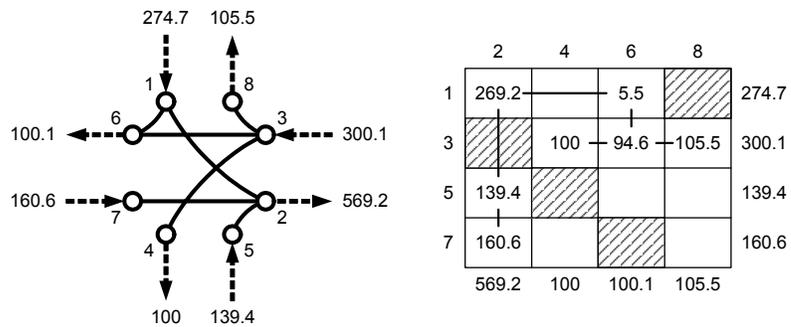
(a) The network and tableau representations of the problem

Figure B.6 The second numerical example and its solutions by the simplex-based algorithm

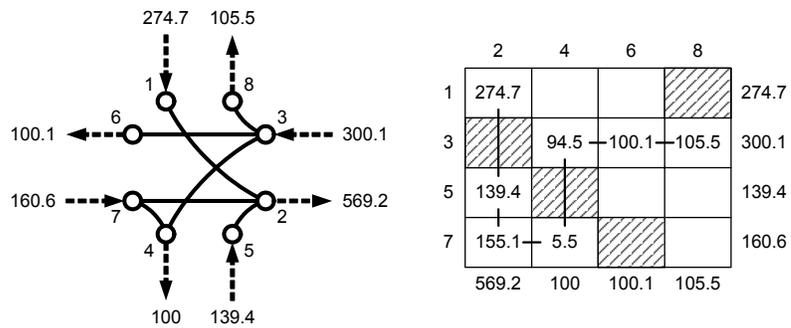
Figure B.6 (Continued)



(b) Iteration 0 (Objective function value: 7)



(c) Iteration 1 (Objective function value: 3)



(d) Iteration 1 (Objective function value: 3)

BIBLIOGRAPHY

- Ahuja, R.K., Magnanti, T.L. and Orlin, J.B. (1993). *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, Upper Saddle River, NJ.
- Ardekani, S.A. and Hobeika, A.G. (1988). "Logistics problems in the aftermath of the 1985 Mexico City earthquake." *Transportation Quarterly*, Vol. 42, No. 1, 107-124.
- Balakrishnan, A. (1984). "Valid inequalities for the network design problem with an application to LTL consolidation problem." Ph.D. Thesis, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA.
- Balakrishnan, A., Magnanti, T.L. and Wong, R.T. (1989). "A dual-ascent procedure for large-scale uncapacitated network design." *Operations Research*, Vol. 37, No. 5, 716-740.
- Bazaraa, M.S., Jarvis, J.J. and Sherali, H.D. (1990). *Linear Programming and Network Flows*. John Wiley & Sons, New York, NY.
- Billheimer, J.W. and Gray, P. (1973). "Network design with fixed and variable cost elements." *Transportation Science*, Vol. 7, No. 1, 49-74.
- Blendon, R.J., Benson, J.M., Buhr, T., Weldon, K.J. and Herrmann, M.J. (2006). "High-risk area hurricane survey." Technical Report, Harvard School of Public Health Project on the Public and Biological Security, Harvard University, Cambridge, MA.
- Boffey, T.B. and Hinxman, A.I. (1979). "Solving the optimal network problem." *European Journal of Operational Research*, Vol. 3, No. 5, 386-393.
- Bower, J.C., Millard, W.D. and Matsumoto, S.W. (1990). *Integrated Emergency Management System (IEMS) User's Manual*. Pacific Northwest Laboratory, Richland, WA.
- Boyce, D.E., Farhi, A. and Weischedel, R. (1973). "Optimal network problem: A branch-and-bound algorithm." *Environment and Planning*, Vol. 5, No. 4, 519-533.
- Boyce, D.E. and Janson, B.N. (1980). "A discrete transportation network design problem with combined trip distribution and assignment." *Transportation Research*, Vol. 14B, No. 1-2, 147-154.
- Boyce, D.E. and Soberanes, J.L. (1979). "Solutions to the optimal network design problem with shipments related to transportation cost." *Transportation Research*, Vol. 13B, No. 1, 65-80.

- Burstable, R.M. (1966). "The computer design of electricity supply networks by a heuristic method." *The Computer Journal*, Vol. 9, No. 3, 263-274.
- Cantarella, G.E., Pavone, G. and Vitetta, A. (2006). "Heuristics for urban road network design: Lane layout and signal settings." *European Journal of Operational Research*, Vol. 175, No. 3, 1682-1695.
- Caudill, R.J. and Kuo, N.M. (1983). "Development of an interactive planning model for contraflow lane evaluation." *Transportation Research Record*, No. 906, 47-54.
- Chan, Y. (1976). "A method to simplify network representation in transportation planning." *Transportation Research*, Vol. 10, No. 2, 179-191.
- Chan, Y., Shen, T.S. and Mahaba, N.M. (1989). "Transportation network design problem: Application of a hierarchical search algorithm." *Transportation Research Record*, No. 1251, 24-34.
- Chen, M. and Alfa, A.S. (1991). "A network design algorithm using a stochastic incremental traffic assignment approach." *Transportation Science*, Vol. 25, No. 3, 215-224.
- Chen, X. and Zhan, F.B. (2008). "Agent-based modeling and simulation of urban evacuation: Relative effectiveness of simultaneous and staged evacuation strategies." *Journal of the Operational Research Society*, Vol. 59, No. 1, 25-33.
- Chien, S.I. and Korikanthimath, V.V. (2007). "Analysis and modeling of simultaneous and staged emergency evacuations." *Journal of Transportation Engineering*, Vol. 133, No. 3, 190-197.
- Church, R.L. and Sexton, R. (2002). "Modeling small area evacuation: Can existing transportation infrastructure impede public safety?" Technical Report, Vehicle Intelligence and Transportation Analysis Laboratory, University of California, Santa Barbara, CA.
- Cova, T.J. and Johnson, J.P. (2002). "Microsimulation of neighborhood evacuations in the urban-wildland interface." *Environment and Planning*, Vol. 34A, No. 11, 2211-2229.
- Cova, T.J. and Johnson, J.P. (2003). "A network flow model for lane-based evacuation routing." *Transportation Research*, Vol. 37A, No. 7, 579-604.
- Crainic, T.G., Gendreau, M. and Farvolden, J.M. (2000). "A simplex-based tabu search method for capacitated network design." *INFORMS Journal on Computing*, Vol. 12, No. 3, 223-236.

- Current, J. and Min, H.K. (1986). "Multiobjective design of transportation networks: Taxonomy and annotation." *European Journal of Operational Research*, Vol. 26, No. 2, 187-201.
- Dial, R.B. (1971). "A probabilistic multipath traffic assignment problem which obviates path enumeration." *Transportation Research*, Vol. 5, No. 2, 83-111.
- Dionne, R. and Florian, M. (1979). "Exact and approximate algorithms for optimal network design." *Networks*, Vol. 9, No. 1, 37-59.
- de Silva, F.N. and Eglese, R.W. (2000). "Integrating simulation modeling and GIS: Spatial decision support systems for evacuation planning." *Journal of Operational Research Society*, Vol. 51, No. 4, 423-430.
- Drezner, Z. and Wesolowsky, G.O. (1997). "Selecting an optimum configuration of one-way and two-way routes." *Transportation Science*, Vol. 31, No. 4, 386-394.
- Drezner, Z. and Wesolowsky, G.O. (2003). "Network design: Selection and design of links and facility location." *Transportation Research*, Vol. 37A, No. 3, 241-256.
- Dunn, C.E. and Newton, D. (1992). "Optimal routes in GIS and emergency planning applications." *Area*, Vol. 24, No. 3, 259-267.
- Erlander, S. (1977). "Accessibility, entropy and the distribution and assignment of traffic." *Transportation Research*, Vol. 12, No. 4, 291-293.
- FEMA. (1984). "Application of the I-DYNEV system to compute estimates of evacuation travel times at nuclear power stations." Federal Emergency Management Agency Report 8, Washington, DC.
- Fisk, C.S. (1980). "Some developments in equilibrium traffic assignment." *Transportation Research*, Vol. 14B, No. 3, 243-255.
- Fisk, C.S. (1984). "Game theory and transportation systems modeling." *Transportation Research*, Vol. 18B, No. 4-5, pp. 301-313.
- Friesz, T.L., Anandalingam, G., Mehta, N.J., Nam, K., Shah, S.J. and Tobin, R.L. (1993). "The multiobjective equilibrium network design problem revisited: A simulated annealing approach." *European Journal of Operational Research*, Vol. 65, No. 1, 44-57.
- Garrison, W.L. and Marble, D.F. (1958). "Analysis of highway networks: A linear programming formulation." *Highway Research Board Proceedings*, No. 37, 1-14.

Gendreau, M. (2002). "An introduction to tabu search." Research Report, Centre de recherche sur les transports, Université de Montréal, Montréal, Québec, Canada.

Gendreau, M., Hertz, A. and Laporte, G. (1994). "A tabu search heuristic for the vehicle routing problem." *Management Science*, Vol. 40, No. 10, 1276-1290.

Gendron, B., Crainic, T.G. and Frangioni, A. (1998). "Multicommodity capacitated network design." B. Sanso and P. Soriano, Eds. *Telecommunications Network Planning*. Luwer, Norwell, MA, 1-19.

Geoffrion, A.M. (1972). "Generalized Benders decomposition." *Journal of Optimization Theory and Applications*, Vol. 10, No. 4, 237-260.

Geoffrion, A.M. (1974). "Lagrangian relaxation for integer programming." *Mathematical Programming Study*, Vol. 2, No. 1, 82-114.

Glickman, T.S. (1970). "Optimal periodic control of reversible traffic operations." Ph.D. Dissertation, Department of Civil Engineering, Johns Hopkins University, Baltimore, M.D.

Glover, F. (1986). "Future paths for integer programming and links to artificial intelligence." *Computers and Operations Research*, Vol. 13, No. 5, 533-549.

Glover, F. (1990). "Tabu search: A tutorial." *Interfaces*, Vol. 20, No. 4, 74-94.

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

Goldblatt, R.B. and Weinisch, K. (2005). "Evacuation planning, human factors, and traffic engineering." *TR News*, No. 238, 13-17.

Golding, D. and Kasperson, R.E. (1988). "Emergency planning and nuclear power: Looking to the next accident." *Land Use Policy*, Vol. 5, No. 1, 19-36.

Haghani, A.E. and Daskin, M.S. (1984). "Network design application of an extraction algorithm for network aggregation." *Transportation Research Record*, No. 944, 37-46.

Haghani, A.E. and Daskin, M.S. (1986). "Aggregation effects on the network design problem." *Journal of Advanced Transportation*, Vol. 20, No. 3, 239-258.

Han, A.F. (1990). "TEVACS: Decision support system for evacuation planning in Taiwan." *Journal of Transportation Engineering*, Vol. 116, No. 6, 821-830.

Hemphill, J. and Surti, V.H. (1974). "A feasibility study of a reversible-lane facility for a Denver street corridor." *Transportation Research Record*, No. 1974, 29-32.

- Hershendorfer, A. (1965). "Optimal utilization and synthesis of road networks." Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, MA.
- Hoang, H.H. (1973). "A computational approach to the selection of an optimal network." *Management Science*, Vol. 19, No. 5, 488-498.
- Hoang, H.H. (1982). "Topological optimization of networks: A nonlinear mixed integer model employing generalized Benders decomposition." *IEEE Transactions on Automatic Control*, Vol. 27, No. 1, 164-169.
- Hobeika, A.G. and Jamei, B. (1985). "MASSVAC: A model for calculating evacuation times under natural disasters." *Proceedings of the Conference on Computer Simulation in Emergency Planning*, Vol. 15, No. 1, 23-28, La Jolla, CA.
- Hobeika, A.G. and Kim, C. (1998). "Comparison of traffic assignments in evacuation modeling." *IEEE Transactions on Engineering Management*, Vol. 45, No. 2, 192-198.
- Hobeika, A.G., Kim, S. and Beckwith, R.E. (1994). "A decision support system for developing evacuation plans around nuclear power stations." *Interfaces*, Vol. 24, No. 5, 22-35.
- Holgún-Veras, J., Pérez, S., Ukkusuri, T., Wachtendorf, T. and Brown, B. (2007). "Emergency logistics issues in Hurricane Katrina: A synthesis and preliminary suggestions for improvement." *Transportation Research Record*. ([In Press](#))
- Hamza-Lup, G.L., Hua, K.A., Le, M. and Peng, R. (2004). "Enhancing intelligent transportation systems to improve and support homeland security." *Proceedings of the 7th IEEE Intelligent Transportation Systems Conference*, Washington, DC, October 3-6, 2004.
- Hamza-Lup, G.L., Hua, K.A. and Peng, R. (2007). "Leveraging e-transportation in real-time traffic evacuation management." *Electronic Commerce Research and Applications*, Vol. 6, No. 4, 413-424.
- Janson, B.N., Buckels, L.S. and Peterson, B.E. (1991). "Network design programming of U.S. highway improvements." *Journal of Transportation Engineering*, Vol. 117, No. 4, 457-478.
- Janson, B.N. and Husaini, A. (1987). "Heuristic ranking and selection procedures for network design problem." *Journal of Advanced Transportation*, Vol. 21, No. 1, 17-46.

- Jeon, K., Lee, J.S., Ukkusuri, S. and Waller, S.T. (2006). "Selectorecombinative genetic algorithm to relax computational complexity of discrete network design problem." *Transportation Research Record*, No. 1964, 91-103.
- Jha, M., Moore, K. and Pashaie, B. (2004). "Emergency evacuation planning with microscopic traffic simulation." *Transportation Research Record*, No. 1886, 40-48.
- Johnson, D.S., Lenstra, J.K. and Rinnooy Kan, A.H.G. (1978). "The complexity of the network design problem." *Networks*, Vol. 8, No. 4, 279-285.
- Johnson, J.H. and Zeigler, D.J. (1983). "Distinguishing human responses to radiological emergencies." *Economic Geography*, Vol. 59, No. 4, 386-402.
- Kwon, E. and Pitt, S. (2005). "Evaluation of emergency evacuation strategies for downtown event traffic using a dynamic network model." *Transportation Research Record*, No. 1922, 149-155.
- Lamar, B.W., Sheffi, W.B. and Powell, W.B. (1990). "A capacity improvement lower bound for fixed charge network design problems." *Operations Research*, Vol. 38, No. 4, 704-710.
- LeBlanc, L.J. (1975). "An algorithm for the discrete network design problem." *Transportation Science*, Vol. 9, No. 3, 183-199.
- LeBlanc, L.J., Morlok, E.K. and Pierskalla, W.P. (1975). "An efficient approach to solving the road network equilibrium traffic assignment problem." *Transportation Research*, Vol. 9, No. 5, 309-318.
- Li, Y., Ziliaskopoulos, A.K. and Waller, S.T. (1999). "Linear programming formulations for system optimum dynamic traffic assignment with arrival time-based and departure time-based demands." *Transportation Research Record*, No. 1667, 52-59.
- Lindell, M.K. and Perry, R.W. (1991). "Understanding evacuation behavior: An editorial introduction." *International Journal of Mass Emergencies and Disasters*, Vol. 9, No. 2, 133-136.
- Liu, Y., Lai, X. and Chang, G.L. (2006a). "Two-level integrated optimization system for planning of emergency evacuation." *Journal of Transportation Engineering*, Vol. 132, No. 10, 800-807.
- Liu, Y., Lai, X. and Chang, G.L. (2006b). "Cell-based network optimization model for staged evacuation planning under emergencies." *Transportation Research Record*, No. 1964, 127-135.

- Liu, Y., Chang, G.L., Liu, Y. and Lai, X. (2008). "Corridor-based emergency evacuation system for Washington D.C.: System development and case study." *Transportation Research Record*. ([In Press](#))
- Los, M. and Lardinois, C. (1982). "Combinatorial programming, statistical optimization and the optimal transportation network problem." *Transportation Research*, Vol. 16B, No. 2, 89-124.
- MacDorman, L.C. (1965). "Case study in sensitivity of highway economic factors." *Highway Research Record*, No. 100, 2-19.
- MacKinnon, R.D. and Hodgson, M.J. (1970). "Optimal transportation networks: A case study of highway systems." *Environment and Planning*, Vol. 2, No. 4, 267-284.
- Magnanti, T.L., Mireault, P. and Wong, R.T. (1986). "Tailoring Benders decomposition for uncapacitated network design." *Mathematical Programming Study*, Vol. 26, No. 2, 112-154.
- Magnanti, T.L. and Wong, R.T. (1984). "Network design and transportation planning: Models and algorithms." *Transportation Science*, Vol. 18, No. 1, 1-55.
- Maher, M.J. (1992). "SAM—A stochastic assignment model." *Mathematics in Transport Planning and Control*, J.D. Griffiths eds., Oxford University Press, Oxford, England, U.K.
- Maher, M.J. and Hughes, P.C. (1997). "A probit-based stochastic user equilibrium assignment model." *Transportation Research*, Vol. 31B, No. 4, pp. 341-355.
- Meitzler, W.D., Martin, P.D., Bower, J.C. and Jaske, R.T. (1986). "Integrated emergency management information system (IEMIS)." *Proceedings of the 1986 Computer Simulation Conference*, Reno, NV.
- Meng, Q., Khoo, H.L. and Cheu, R.L. (2008). "Microscopic traffic simulation model-based optimization approach for the contraflow lane configuration problem." *Journal of Transportation Engineering*, Vol. 134, No. 1, 41-49.
- Minoux, M. (1989). "Network synthesis and optimum network design problems: Models, solution methods and applications." *Networks*, Vol. 19, No. 3, 313-360.
- Mouskos, K.C. "A tabu-based heuristic search strategy to solve a discrete transportation equilibrium network design problem." Ph.D. Thesis, Department of Civil, Environmental and Architectural Engineering, the University of Texas, Austin, TX.

Murray-Tuite, P.M. and Mahmassani, H.S. (2004). "Transportation network evacuation planning with household activity interactions." *Transportation Research Record*, No. 1894, 150-159.

Ochoa-Rosso, F. (1968). "Applications of discrete optimization techniques to capital investment and network synthesis problems." Research Report No. R68-42, Department of Civil Engineering, MIT, Cambridge, MA.

Parker, R.G. and Rardin, R.L. (1988). *Discrete Optimization*. Academic Press, New York, NY.

Pearman, A.D. (1974). "Heuristic approaches to road network optimization." *Engineering Optimization*, Vol. 1, No. 1, 37-49.

Pidd, M. de Silva, F.N. and Eglese, R.W. (1996). "A simulation model for emergency evacuation." *European Journal of Operational Research*, Vol. 90, No. 3, 413-419.

Pidd, M., Eglese, R.W. and de Silva, F.N. (1997). "CEMPS: A prototype spatial decision support system to aid in planning emergency evacuations." *Transactions in GIS*, Vol. 1, No. 4, pp. 321-334.

Poorzahedy, H. and Turnquist, M.A. (1982). "Approximate algorithms for the discrete network design problem." *Transportation Research*, Vol. 16B, No. 1, 45-55.

Radwan, A.E., Hobeika, A.G. and Sivasailam, D. (1985). "A computer simulation model for rural network evacuation under natural disasters." *ITE Journal*, Vol. 55, No. 9, pp. 25-30.

Rathi, A.K. and Solanki, R.S. (1993). "Simulation of traffic flow during emergency evacuations: A microcomputer based modeling system." *Proceedings of the 1993 Winter Simulation Conference*, 1250-1258, Los Angeles, CA.

Ridley, T.M. (1968). "An investment policy to reduce the travel time in a transportation network." *Transportation Research*, Vol. 2, No. 4, 409-424.

Rothengatter, W. (1979). "Application of optimal subset selection to problems of design and scheduling in urban transportation networks." *Transportation Research*, Vol. 13B, No. 1, 49-63.

Scott, A.J. (1967). "A programming model of an integrated transportation network." *Papers of the Regional Science Association*, Vol. 19, No. 4, 215-222.

Scott, A.J. (1969). "The optimal network problem: Some computational procedures." *Transportation Research*, Vol. 3, No. 3, 201-210.

- Sheffi, Y. (1985). *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Prentice Hall, Englewood, NJ.
- Sheffi, Y., Mahmassani, H. and Powell, W.B. (1981). "Evacuation studies for nuclear power plant sites: A new challenge for transportation engineers." *ITE Journal*, Vol. 51, No. 6, 25-28.
- Sheffi, Y., Mahmassani, H. and Powell, W.B. (1982). "A transportation network evacuation model." *Transportation Research*, Vol. 16A, No. 3, 209-218.
- Sheffi, Y. and Powell, W.B. (1982). "An algorithm for equilibrium assignment problem with random link times." *Networks*, Vol. 12, No. 2, 191-207.
- Shekhar, S. and Kim, S. (2006). "Contraflow transportation network reconfiguration for evacuation route planning." CTS Project Report, Minnesota Department of Transportation, St. Paul, MS.
- Sherali, H.D., Carter, T.B. and Hobeika, A.G. (1991). "A location-allocation model and algorithm for evacuation planning under hurricane/flood conditions." *Transportation Research*, Vol. 25B, No. 6, 439-452.
- Sinuany-Stern, Z. and Stern, E. (1993). "Simulating the evacuation of a small city: The effects of traffic factors." *Socio-Economic Planning Sciences*, Vol. 27, No. 2, 97-108.
- Sivanandan, R., Hobeika, A.G., Ardekani, S.A. and Lockwood, P.B. (1988). "A heuristic shortest-path method for emergency vehicle assignment—a study on the Mexico City network." *Transportation Research Record*, No. 1168, 86-91.
- Sorensen, J.H., Vogt, B.M. and Mileti, D.S. (1987). *Evacuation: An Assessment of Planning and Research*. Technical Report ORNL-6376, Oak Ridge National Laboratory, Oak Ridge, TN.
- Southworth, F. and Chin, S.-M. (1987). "Network evacuation modeling for flooding as a result of dam failure." *Environment and Planning*, Vol. 19A, No. 11, 1543-1558.
- Stairs, S. (1968). "Selecting an optimal traffic network." *Journal of Transport Economics and Policy*, Vol. 2, No. 3, 218-231.
- Steenbrink, P.A. (1974a). *Optimization of Transport Networks*. John Wiley & Sons, New York, NY.
- Steenbrink, P.A. (1974b). "Transport network optimization in the Dutch integral transportation study." *Transportation Research*, Vol. 8B, No. 1, 11-27.

Stern, E. and Sinuany-Stern, Z. (1989). "A behavioral-based simulation model for urban evacuation." *Papers of the Regional Science Association*, Vol. 66, No. 1, 87-103.

Theodoulou, G. and Wolshon, B. (2004). "Alternative methods to increase the effectiveness of freeway contraflow evacuation." *Transportation Research Record*, No. 1865, 48-56.

Tufekci, S. and Kisko, T.M. (1991). "Regional evacuation modeling system (REMS): A decision support system for emergency area evacuation." *Computers and Industrial Engineering*, Vol. 21, No. 1, 89-93.

Tung, R.S.I. and Schneider, J.B. (1987). "Designing optimal transportation networks: An expert systems approach." *Transportation Research Record*, No. 1145, 20-27.

Tuydes, H. and Ziliaskopoulos, A.K. (2006). "Tabu-based heuristic approach for optimization of network evacuation contraflow." *Transportation Research Record*, No. 1964, 157-168.

Urbina, E. and Wolshon, B. (2003). "National review of hurricane evacuation plans and policies: A comparison and contrast of state practices." *Transportation Research*, Vol. 37A, No. 3, 257-275.

Wei, C.H. and Schonfeld, P.M. (1993). "An artificial neural network approach for evaluating transportation network improvements." *Journal of Advanced Transportation*, Vol. 27, No. 2, 129-151.

Wolshon, B. (2001). "One-way-out: Contraflow freeway operation for hurricane evacuation." *Natural Hazards Review*, Vol. 2, No. 3, 105-112.

Wolshon, P.B. and Lambert, L. (2004). *Convertible Roadways and Lanes: A Synthesis of Highway Practice*. NCHRP Synthesis 340, National Cooperative Highway Research Program, Transportation Research Board, Washington, D.C.

Wolshon, B., Urbina, E., Wilmot, C. and Levitan, M. (2005). "Review of policies and practices for hurricane evacuation I: Transportation planning, preparedness, and response." *Natural Hazards Review*, Vol. 6, No. 3, 129-142.

Wong, R.T. (1980). "Worst-case analysis of network design problem heuristics." *SIAM Journal of Algebraic and Discrete Methods*, Vol. 1, No. 1, 51-63.

Wong, R.T. (1985a). "Probabilistic analysis of a network design problem heuristic." *Networks*, Vol. 15, No. 4, 347-363.

Wong, R.T. (1985b). "Transportation network research: Algorithmic and computational questions." *Transportation Research*, Vol. 19A, No. 5/6, 436-438.

Xiong, Y. and Schneider, J.B. (1992). "Transportation network design using a cumulative genetic algorithm and neural networks." *Transportation Research Record*, No. 1364, 37-44.

Xiong, Y. and Schneider, J.B. (1995). "Processing of constraints in transportation network design problem." *Journal of Computing in Civil Engineering*, Vol. 9, No. 1, 21-28.

Yamada, T. (1996). "A network flow approach to a city emergency evacuation planning." *International Journal of Systems Science*, Vol. 27, No. 10, 931-936.

Yang, H. and Bell, M.G.H. (1998). "Models and algorithms for road network design: A review and some new developments." *Transport Review*, Vol. 18, No. 3, 257-278.

Yuan, F., Han, L.D., Chin, S.-M. and Hwang, H. (2006). "Proposed framework for simultaneous optimization of evacuation traffic destination and route assignment." *Transportation Research Record*, No. 1964, 50-58.

Ziliaskopoulos, A.K. (2000). "A linear programming model for the single destination system optimum dynamic traffic assignment problem." *Transportation Science*, Vol. 34, No. 1, 37-49.

Ziliaskopoulos, A.K. and Waller, S.T. (2000). "An Internet-based geographic information system that integrates data, models and users for transportation applications." *Transportation Research*, Vol. 8C, No. 1-6, 427-444.

Zou, N., Yeh, S.T., Chang, G.L., Marques, A. and Zezeski, M. (2005). "Simulation-based emergency evacuation system for Ocean City, Maryland, during hurricanes." *Transportation Research Record*, No. 1922, 138-148.